GPIB and IQ-Application Development Guide

EXFO

IQ-200

GPIB and IQ-Application Development Guide

April 1996

P/N: MAN-075-I .1AN

First Edition

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system or transmitted, in any form by any means, without the prior written permission of EXFO Electro-Optical Engineering Inc. (EXFO).

Information furnished by EXFO is believed to be accurate and reliable. However, no responsibility is assumed by EXFO for its use, nor for any infringements of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent rights of EXFO.

EXFO's Commerce And Government Entities (CAGE) code under the North Atlantic Treaty Organization (NATO) is 0L8C3.

The information contained in this document is subject to change without notice.

© 1996 EXFO Electro-Optical Engineering Inc.

Table of Contents

Page

1 INTRODUCTION 1-1

1.1 Presentation 1-1
1.2 Conventions 1-2
1.3 List of Abbreviations 1-3
1.4 Transportation and Storage 1-3
1.5 Getting Help 1-4

2 IQ-200 INTEGRATED APPLICATIONS - OVERVIEW 2-1

2.1 Introduction 2-1
2.2 GPIB Device 2-1
2.3 Integrated Applications Using DDE 2-3
2.4 GPIB Controller 2-5
2.5 The IQ-206 PC Expansion Card with the IQ-206 2-7

3 SCPI MANAGER 3-1

3.1 Purpose 3-1
3.2 Starting the SCPI Manager 3-2
3.3 Configuring the SCPI Manager 3-5
3.4 Operating Modes 3-7
3.5 Viewing Register Data 3-8

4 THE IQ-203 AS A GPIB DEVICE 4-1

4.1 GPIB Capability 4-1
4.2 Configuration 4-2
4.3 Getting Started 4-2
4.4 Establishing GPIB Communication 4-3
4.5 Programming 4-4
4.6 Driver Packages 4-4

5 INTEGRATED APPLICATIONS (DDE) 5-1

5.1 Introduction 5-1
5.2 Background Information 5-2
5.3 Opening a DDE Communication Channel 5-5

Page

5.4 Useful Programming Hints 5-6
5.5 Sample VB Program 5-8

6 THE IQ-203 AS A GPIB CONTROLLER 6-1

6.1 Introduction 6-1
6.2 Supplied Hardware and Software 6-1
6.3 Configuration 6-2
6.4 Programming 6-2

7 SCPI COMMANDS 7-1

7.1 Standard Status Data Structure 7-1
7.2 Common Commands 7-3
7.3 System Commands 7-10
7.4 IQ-200 SCPI Command Structure 7-11
7.4.1 Module SCPI Command Address 7-12
7.5 IQ-200 Optical Test System Commands 7-14
7.6 IQ Power Meter Module Commands 7-21
7.7 IQ-3200 Return Loss Meter Module Commands 7-35
7.8 IQ-2100 Source Commands 7-45
7.9 IQ-3100/3300 Variable Attenuator/Reflector Commands 7-49
7.10 IQ-9100 Switch Module Commands 7-60
7.11 Quick Reference Command Trees 7-66
7.11.1 Optical Test System 7-66
7.11.2 IQ-1100 and IQ-1200 Power Meters 7-67
7.11.3 IQ-2100 Source 7-69
7.11.4 IQ-3200 Return Loss Meter 7-70
7.11.5 IQ-3100 Variable Attenuator / IQ-3300 Variable Reflector 7-71
7.11.6 IQ-9100 Optical Switches 7-72

8 ERROR MESSAGES 8-1

8.1 Error Message Format 8-1
8.2 SCPI Manager Errors 8-2
8.3 IQ-200 Optical Test System Error Messages 8-3
8.4 IQ-1100, IQ-1200, and IQ-1500 Power Meter Error Messages 8-4
8.5 IQ-2100 Source Error Messages 8-5
8.6 IQ-3200 Return Loss Meter Error Messages 8-5
8.7 IQ-3100 and IQ-3300 Error Messages 8-6

Page

8.8 IQ-9100 Switch Error Messages 8-7

9 GENERAL INFORMATION 9-1

9.1 Warranty 9-1
9.2 Liability 9-2
9.3 Exclusions 9-2
9.4 Certification 9-2
9.5 Service and Repairs (Under Warranty or Other) 9-3

10 INDEX 10-1

Figures and Tables

Page

Figure 2-1. GPIB Device Configuration 2-2
Figure 2-2. IQ Integrated Applications 2-4
Figure 2-3. GPIB Controller Configuration 2-6
Figure 2-4. IQ-206 with PC Expansion Card 2-8
Figure 3-1. Starting the SCPI Manager 3-3
Figure 3-2. SCPI Manager Window 3-4
Figure 3-3. SCPI Manager Communications Parameters Window 3-6
Figure 3-4. Viewing Standard Registers 3-8
Figure 5-1. DDE Communication with SCPI Manager 5-2
Figure 7-1. Standard Status Data Structures (IEEE 488.2) 7-2
Figure 7-2. IQ-200 Module Addresses 7-13
Figure 8-1. Error Message Format 8-1

Table 4-1. IQ-203 GPIB Device Capability 4-1
Table 7-1. Standard Registers 7-1
Table 7-2. Common Command Summary 7-3

1 INTRODUCTION

1.1 Presentation

The IQ-200 General Purpose Interface Bus (GPIB) and IQ-Application Development Guide offers a comprehensive introduction into the various IQ configurations. This document also provides detailed information that will enable a programmer to create integrated applications, controlling the IQ-200 system components either locally or through an IEEE 488 (GPIB) instrument bus.

The following subjects are discussed.

Integrated Applications - Overview

This chapter describes the different combinations for which the IQ-200 system may be configured. This overview covers a number a important concepts, including how the various hardware and software components interact.

SCPI Manager

This chapter describes the Standard Commands for Programmable Instruments (SCPI) Manager, a simple but very important program that provides the interface between the IQ-200 system and the integrated applications. Understanding the SCPI Manager is very important, whether programming for GPIB or Dynamic Data Exchange (DDE) control.

GPIB Controller, GPIB Device, and DDE

These three chapters describe how to configure and operate the IQ-200 components with each of the different interface options. Where applicable, example programs or sections of code are included.

SCPI Commands

This chapter provides an introduction into the SCPI standard, describes the IQ-200 SCPI command format, and provides a detailed description of all the available commands. At the end of the chapter, quick reference command trees for each module are also provided.

Error Messages

This chapter describes the error message format, followed by a list of all the error messages.

1.2 Conventions

WARNING, CAUTION, and IMPORTANT

The following conventions should be clearly understood before operating the unit:

WARNING Refers to a potential personal hazard. It requires a procedure which, if not correctly followed, may result in bodily harm or injury. Do not proceed beyond a **WARNING** unless the required conditions are fully understood and met.

CAUTION Refers to a potential product hazard. It requires a procedure which, if not correctly followed, may result in irreparable component damage. Do not proceed beyond a **CAUTION** unless the required conditions are fully understood and met.

IMPORTANT Refers to any information regarding the operation of the product which should not be overlooked.

1.3 List of Abbreviations

DMA	Direct Memory Addressing
DDE	Dynamic Data Exchange
DLL	Dynamic Link Library
EOI	End of Image Marker
EOS	Effective Opening Size
ESE	Standard Event Status Enable Register
ESR	Standard Event Status Register
FIFO	First In First Out
GPIB	General Purpose Interface Bus
ISA	Industry Standard Architecture
SCPI	Standard Commands for Programmable Instruments
SRE	Service Request Enable Register
SRQ	Service Request
STB	Status Byte Register

1.4 Transportation and Storage

Maintain a temperature range within specifications when transporting or storing the unit. Transportation damage can occur from improper handling. Therefore, the following steps are recommended to minimize the possibility of damage.

- If the unit must be shipped, pack it in the original packing material for best protection.
- When not in use, store unit at room temperature in a clean and dry area. Avoid high humidity and wide temperature fluctuations.
- Keep the unit out of direct sunlight.
- Avoid unnecessary shocks and vibrations.

1.5 Getting Help

If you encounter any difficulty while operating the unit, please call EXFO at one of the offices below. The Sales and Product Support Group is available from 7:30 a.m. to 8:00 p.m. eastern time, Monday to Friday. Any sales offices can also be contacted if questions or problems arise.

Sales and Product Support Group

- USA and Canada

  Tel.: 1-800-663-EXFO
  1-800-663-3936
  Fax: 1-418-683-2170
  E-Mail: exfo@riq.qc.ca

- International

  Tel.: 1-418-683-0211
  Fax: 1-418-683-2170
  E-Mail: exfo@riq.qc.ca

EXFO Europe

Tel.: 33-1-34.63.00.20
Fax: 33-1-34.65.90.93
E-Mail: exfoeur@ibm.net

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

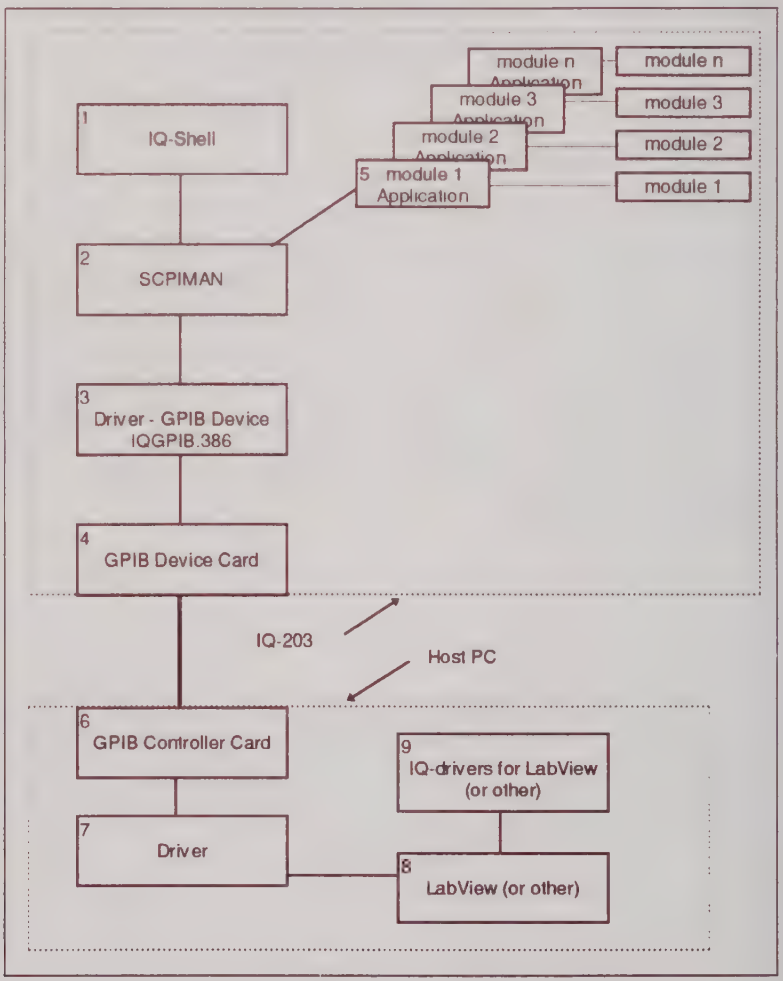

Figure 2-1. GPIB Device Configuration

CAUTION

The GPIB Device card must be installed at the factory. Attempting to install any other GPIB card will void the product warranty.

In this configuration the GPIB Controller card (available from several manufacturers) and application software (provided by the user) are installed in the host PC. Commands will be transmitted along the GPIB chain, recognized and decoded by the IQ-203 GPIB Device card (4), interpreted by the SCPI Manager (2) and dispatched to the appropriate module control software (5). The SCPI Manager, fully explained in Chapter 3, is a standard component of the IQ software package.

Applications can be created in any programming environment capable of communicating with a GPIB compatible instrument. For the common commercial packages (e.g. LabVIEW) software drivers are available.

See Chapter 4 *The IQ-203* as a *GPIB Device*, for detailed information about this configuration.

2.3 Integrated Applications Using DDE

The IQ-203 mainframe is built around a DOS/Windows PC platform. When operating a module as a stand-alone instrument a Windows application controls the module. An interface screen is displayed when the module front panel LED push button is pressed.

Windows Dynamic Data Exchange (DDE) provides a communication link permitting bidirectional data sharing between two applications. The block diagram illustrated in Figure 2-2 will help to explain this configuration.

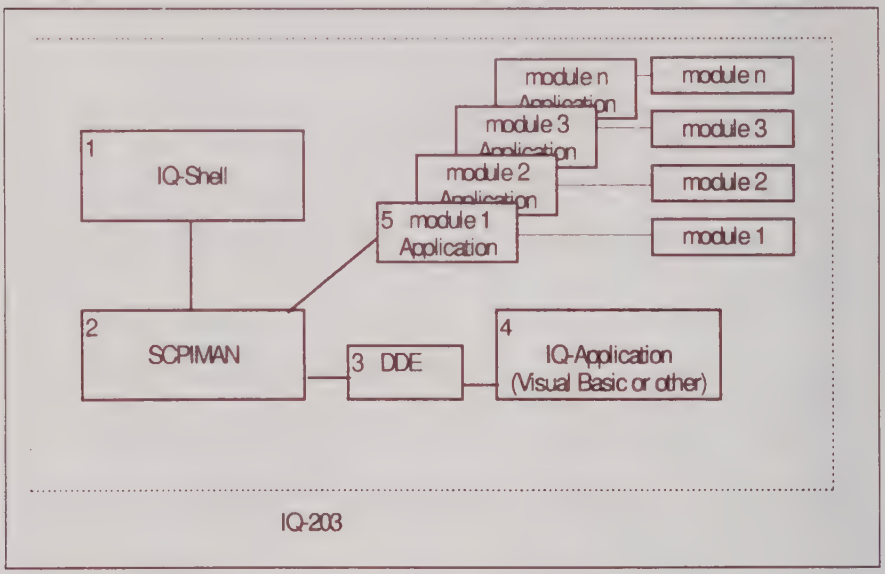

Figure 2-2. IQ Integrated Applications

With reference to Figure 2-2, the integrated IQ-Application (4) must open a DDE link (3) with the SCPI Manager (2). Once this link is open, commands to the modules are routed by the SCPI Manager to the appropriate module control software. Data from the modules is routed through the SCPI Manager and returned back to the calling application.

Integrated IQ-Applications may be created using any programming environment that supports Windows 3.1 DDE.™ Visual Basic™, Delphi™, C++™, LabVIEW™, and LabWindows/CVI™ all support this communication protocol. Virtual Instrument drivers (for DDE) are being developed for National

Instruments LabVIEW software. Call EXFO for the latest information regarding the availability of these drivers.

Integrated IQ-Applications written at EXFO and contained in the IQ Application Library are created using Visual Basic 4.0, professional edition.

2.4 GPIB Controller

If a GPIB Controller card is installed in the IQ-203 mainframe, applications integrating IQ-200 system components and GPIB compatible instruments can be installed and executed on the IQ-203. This configuration, shown in Figure 2-3, is known as the IQ-203 Controller configuration.

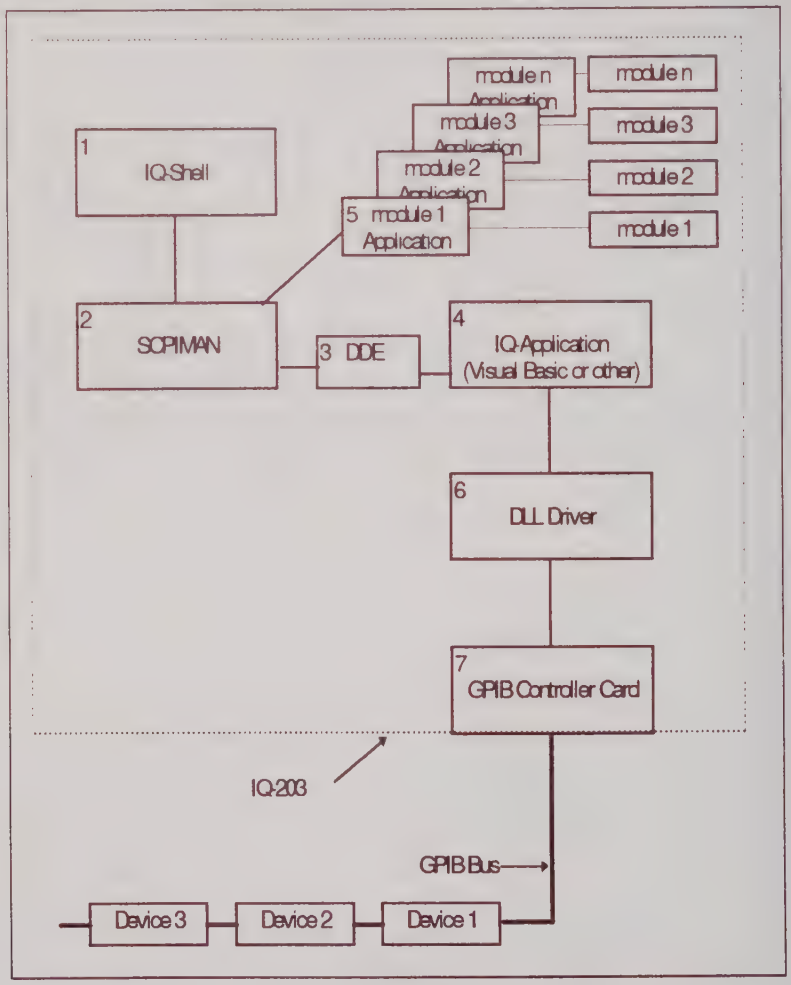

Figure 2-3. GPIB Controller Configuration

CAUTION

The IQ-203 GPIB Controller card must be installed at the factory. Attempting to install any other GPIB card will void the product warranty.

In this particular setup, the IQ performs the role of system controller, controlling the IQ modules (5) by a DDE link (3) and controlling the external GPIB chain using the National Instruments Dynamic Link Library (DLL) driver (6) and controller card (7).

Applications can be written using any programming environment that supports both DDE and GPIB communication. GPIB development kits are available for all of the major commercial programming languages.

2.5 The IQ-206 PC Expansion Card with the IQ-206

The IQ-206 PC Expansion Card can be installed in any PC AT compatible computer with an empty Industry Standard Architecture (ISA) expansion slot. This card offers the option of connecting an IQ-206 (up to a maximum of 4) to a host computer. A block diagram for this configuration is shown in Figure 2-4.

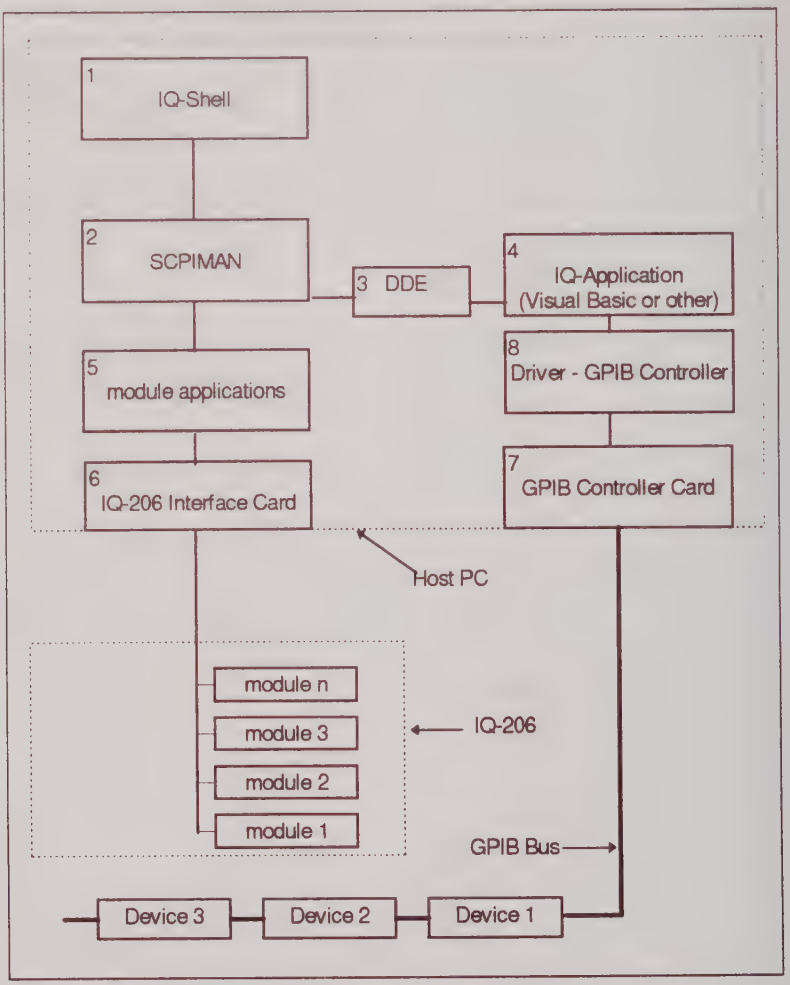

Figure 2-4. IQ-206 with PC Expansion Card

The IQ-206 PC Expansion Card with the IQ-206

The IQ-206 PC Expansion Card and IQ software, along with the host computer, emulate the IQ-203 mainframe. Integrated IQ-Applications can be created and executed on the host PC. The application (4) communicates with the SCPI Manager (2) through a DDE link (3), which in turn communicates with the IQ module control programs (5) and through the interface card (6) to the modules installed in the IQ-206 Expansion Unit

With appropriate software and a GPIB controller card installed in the host computer, this setup also permits control of an external GPIB instrument bus. In this case, the application communicates with the IQ modules through the DDE and communicates with the GPIB devices using the GPIB controller interface.

***Note**: In this configuration, any modules installed in the IQ-206 are not GPIB instruments. In order to control IQ modules you must use a DDE communication link.*

See Chapter 5, *Integrated Applications (DDE)*, for detailed information about developing integrated applications for the IQ-200 system components.

See the documentation supplied with your GPIB controller interface card (not supplied by EXFO for this configuration) for information about programming and controlling GPIB instruments.

3 SCPI MANAGER

3.1 Purpose

The IQ-200 family components are designed to comply with the IEEE-488 and Standard Commands for Programmable Instruments (SCPI) standards. The SCPI standards, more fully explained later in this manual, define a structure for SCPI and makes it possible to develop compatible and efficient integrated test and measurement software. This allows a programmer the flexibility to integrate the operation and measurements of different IQ-200 system modules into a single software application. Such an application greatly simplifies complex measurement procedures and reduces manual intervention, thus improving the overall efficiency of the instrument package.

The SCPI Manager is the software interface that provides the link between the IQ system components and any application that supports either a GPIB interface or a Windows Dynamic Data Exchange (DDE) link. Here are the main functions of the SCPI Manager:

- maintain communication with the various components of the IQ system,
- keep track of errors and events,
- validate commands,
- route commands to the appropriate system component,
- emulate the standard IEEE-488 registers when in DDE mode.

IMPORTANT

The SCPI Manager is designed to fully support the IEEE 488.2 standard for both GPIB and DDE type interfaces. In the DDE mode the SCPI Manager

fully emulates all the standard registers and common commands outlined in Chapter 7, SCPI Commands.

The following important concepts regarding the SCPI Manager should be understood:

- The SCPI Manager communicates with the program that controls the module (i.e. the application that is normally used to control a module), and not directly with the module itself.
- The SCPI Manager can open only one communication link at a time. Simultaneous communication with both GPIB and DDE is not supported.

There are command trees for the IQ-200 Optical Test System (the IQ operating system) and for each of the different IQ modules. These commands are described in detail later in this manual. Complex and repetitive operations can be easily performed by integrating commands into a high level application (GPIB or DDE) and communicating with the SCPI Manager.

3.2 Starting the SCPI Manager

In order to control the IQ modules with the GPIB or DDE interface the SCPI Manager application must be open.

The SCPI Manager is started by selecting **SCPI Manager** from the **Run** menu in the IQ-200 Optical Test System, as shown in Figure 3-1.

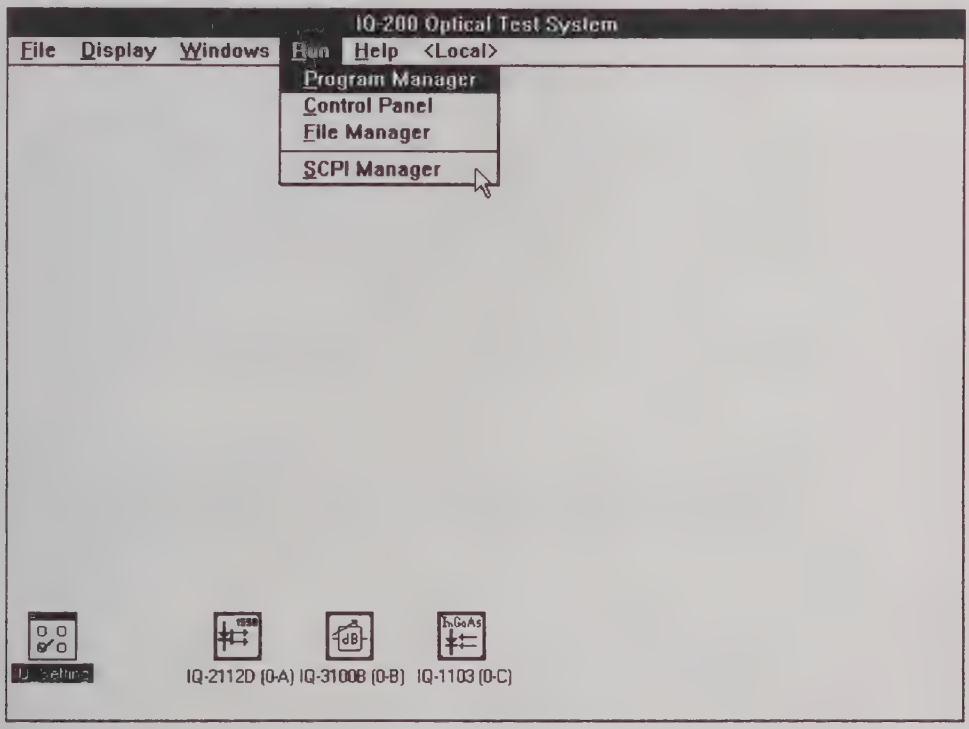

Figure 3-1. Starting the SCPI Manager

The SCPI Manager window displaying the current configuration parameters (Figure 3-2) is displayed.

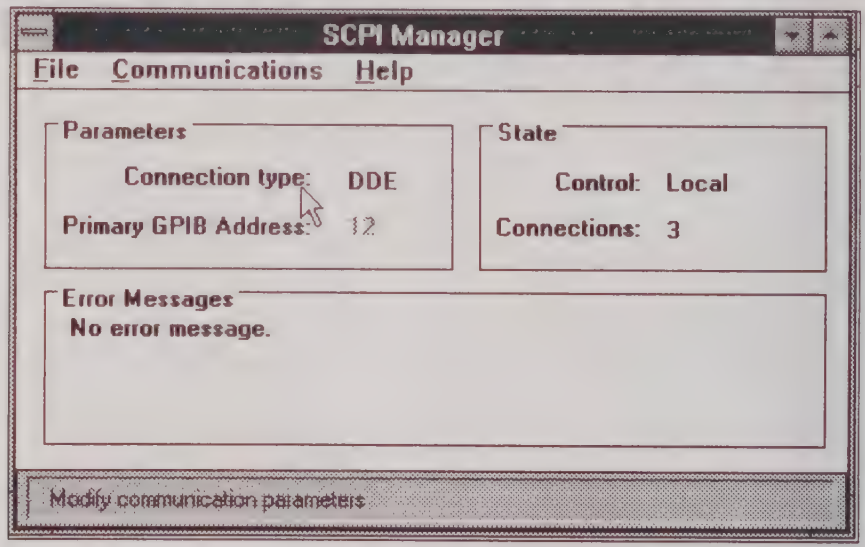

Figure 3-2. SCPI Manager Window

The **Parameters** box displays the current connection type, **DDE** or **GPIB**, and if applicable the current GPIB address. If the connection type is DDE, the GPIB address is not applicable and is displayed in light gray.

The **State** box indicates the current control status, [**Local**, **Remote**, **LocaL Lockout** or **Remote Lockout**.] See Section 3.4, *Operating Modes* for a complete description of the different modes.

The number of active connections is also displayed. This value refers to the number of IQ instruments (modules and the IQ-200 Optical Test System) currently in communication with the SCPI Manager.

***Note**: The SCPI Manager is always in communication with the IQ-200 Optical Test System.*

An IQ module is in communication with the SCPI Manager if the software application for that module is open. Module applications are opened by pressing the module's LED push-button, by double-clicking the module icon, or by sending the appropriate command to the IQ-200 Optical Test System. The SCPI Manager window also displays current and past error messages.

***Note**: The state of the SCPI Manager (open or closed) is saved when saving the Optical Test System configuration. This means that when opening such a configuration, the SCPI Manager will open automatically.*

3.3 Configuring the SCPI Manager

As previously pointed out, in order to remotely control the IQ system components the SCPI Manager must be open. This does not mean that the SCPI Manager window will always be visible on the IQ system screen. Most of the time the SCPI Manager runs in the background or is minimized. To change any of the SCPI parameters, the ALT-TAB (keep the ALT key depressed) key combination should be pressed until the SCPI Manager icon appears in the middle of the screen. Releasing the ALT key at this point will bring the SCPI Manager window to the foreground. In order to do this same step using a mouse, you must be in the IQ-200 Optical Test System screen. Select one of the options from the **Run** menu. For example, highlight **Control Panel** from the three options that will appear, then, activate the Control Menu; choose **Switch To** and then select SCPI Manager.

The following parameters can be modified from within the SCPI Manager;

- Connection type, **GPIB** or **DDE**
- GPIB address, **1** to **15**

Both of these parameters are selected from the **Communication Parameters** window, accessed by selecting Parameters from the SCPI Manager Communication menu. The Communication Parameters window is shown in Figure 3-3.

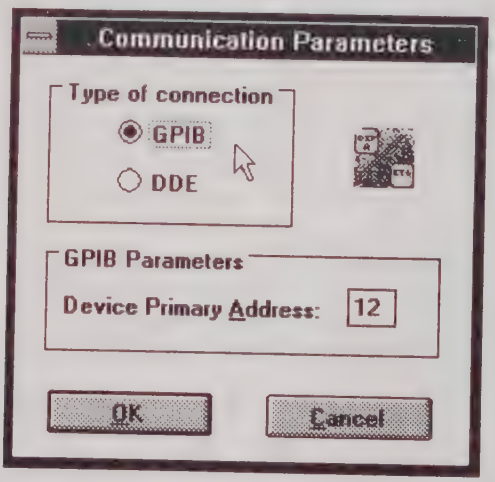

Figure 3-3. SCPI Manager Communications Parameters Window

To change the connection type,

- click on the appropriate **Type of connection** radio button.

***Note**: The GPIB connection is used when a GPIB Device card is installed in the IQ-203. If a GPIB Device card is present, GPIB will be the default communication type.*

To change the GPIB address;

- Enter the desired GPIB address in the edit box. Any address from 1 to 15 may be entered. The default address is 12. The required address is determined by the number of different instruments on the bus. Each GPIB instrument on the same bus must have a unique address.

3.4 Operating Modes

Following the IEEE-488 standard, the SCPI Manager supports three operating modes; **Local**, **Remote**, **Local Lockout** and **Remote Lockout**. The operating mode is sent to and displayed by each module application.

Local mode is the normal operating mode where a module is controlled from its stand-alone software application.

Remote mode indicates that the SCPI Manager has been receiving GPIB commands from an external GPIB controller or from a high level application DDE linked to the SCPI Manager. Remote mode does permit the user to resume local control. A subsequent remote command will then put the IQ system back in Remote mode.

Local Lockout indicates that a high level local application (using DDE) is controlling the SCPI Manager and has enabled the lockout mode. Manual operations are not permitted. See Section 5.4, *Useful Programming Hints*, for information on how to use lockout mode with a DDE linked application.

Remote Lockout indicates that a remote GPIB application is controlling the SCPI Manager and has enabled the lockout mode. Manual operations are not permitted. See GPIB controller documentation for information on how to enable GPIB lockout. (RWLS - remote with lockout state)

3.5 Viewing Register Data

The contents of the four primary status and enable registers, Events Enable, Standard Events, Service Enable, and Status Byte can be viewed by selecting **Registers** from the **Communication** menu. The screen is similar to the one shown in Figure 3-4.

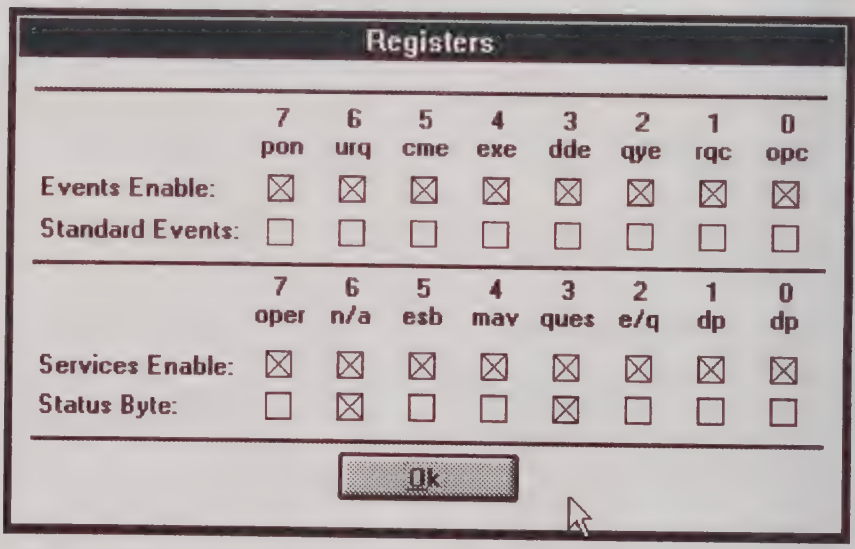

Figure 3-4. Viewing Standard Registers

The significance of each bit in these registers is explained in Section 7.1, *Standard Status Data Structure*. This information may be useful when debugging or troubleshooting.

4 THE IQ-203 AS A GPIB DEVICE

4.1 GPIB Capability

The IQ-203 GPIB Device configuration complies to the IEEE 488.2 (1992) standard to the extent shown in Table 4-1 below.

CODE	FUNCTION	CAPABILITY
SH1	Source Handshake	Complete capability
AH1	Acceptor Handshake	Complete capability
T6	Talker Interface	Basic Talker, Serial Poll, Unaddress if MLA, No Talk Only
L4	Listener Interface	Basic Listener, Unaddress if MTA, No Listen Only
SR1	Service Request	Complete capability
RL1	Remote Local	Complete capability
PP0	Parallel Poll	No capability
DC1	Device Clear	Complete capability
DT1	Device Trigger	Accepted but ignored
C0	Controller	No capability
E2		Three state output and open collector handshake lines

Table 4-1. IQ-203 GPIB Device Capability

4.2 Configuration

The following information describes the GPIB interface driver (IQGPIB.386).

- Default GPIB address **12**
- Maximum number of IQ-203 on GPIB bus **15**
- Input buffer size (circular) **1536 bytes (1.5K)**
- Output buffer size (circular) **512 bytes (0.5K)**
- Maximum number of messages - Input **20**
- Maximum number of messages - Output **20**
- Message termination character (EOS) **0x0A (line feed)**
- Number of bits used - one character **8**
- Generate EOI on sending EOS **active**
- Use of EOS after messages **Input, Output**
- Command transfer delay **None**

4.3 Getting Started

In order to remotely control the IQ-203 from a host computer the following items are required:

- IQ-203 c/w option -I1 (GPIB Device Card)
- Host computer with a GPIB controller card
- GPIB interface cable with standard GPIB Connector (P/N GP-130 available from EXFO)
- GPIB application software installed in the host computer

4.4 Establishing GPIB Communication

To initiate GPIB communication with the IQ-203;

1. Connect the IQ-203 to the GPIB bus.
2. Power on the IQ-203.
3. From the IQ-200 Optical Test System **Run** menu, select **SCPI Manager**.
4. As explained in Section 3.3, *Configuring the SCPI Manager*, set the communication type to **GPIB** and select the appropriate GPIB address. The default values are; communication type = GPIB, and GPIB address = 12

The IQ-203 is now ready to receive remote instructions.

IMPORTANT

As previously mentioned, the SCPI Manager does not communicate directly with the IQ modules. The SCPI Manager communicates with the executable file (*.exe) that controls a module. For this reason, a module application must be open (may be in background or minimized) when using GPIB control. Module applications may be opened manually (press the LED push-button on the front of the module) or remotely (by sending the INST:STAT command to the IQ-200 Optical Test System).

4.5 Programming

Complex measurement programs may be written using any programming environment that supports GPIB communication. GPIB development kits are available for most of the popular commercial programming languages. The IQ-200 optical test system offers many commands permitting complete remote control of all the IQ components. These commands adhere to the SCPI standard and are fully explained in Chapter 7, *SCPI Commands*.

4.6 Driver Packages

Drivers for many commercial GPIB development environments are available (or will soon be) for each IQ component. These drivers provide a virtual instrument interface facilitating remote operation of the IQ system and its modules. The drivers are supplied on a separate disk with each module and must be installed in the appropriate directory on the host (controlling) computer. See the user manual of the GPIB development environment for instructions on how to install and use the IQ Virtual Instrument (VI) drivers. For example, if you are working in the GPIB development environment LabVIEW, you should consult its user manual.

Note: *Virtual Instrument drivers are under development. Contact EXFO for up- to-date information.*

5 INTEGRATED APPLICATIONS (DDE)

5.1 Introduction

As mentioned in the overview in Section 2.3, *Integrated Applications Using DDE*, applications can be developed in any programming environment supporting Windows Dynamic Data Exchange (DDE). Applications controlling one or many modules can be created. This offers the user enormous flexibility for customizing a particular module or creating complex custom measurement procedures.

DDE is a mechanism supported by most Windows applications that allows two programs to exchange data. A DDE link requires that one application act as server (source) and the other application act as client (destination). As far as the IQ is concerned, the SCPI Manager is the DDE server and the integrated application is the DDE client.

Regardless of the programming environment, commands in the application are sent to the SCPI Manager, interpreted, and dispatched to the appropriate module control software. The response to the commands is returned to the integrated application.

IMPORTANT

It is important to understand that the SCPI Manager does not communicate directly with the IQ modules. The SCPI Manager communicates with the executable file. For example, IQPM.exe for the IQ-1100 power meter that normally controls the module. For this reason, the module application must be open when remote controlling an IQ module.

Understanding the SCPI standard and the IQ SCPI command format is critical when developing integrated applications. The command format and the specific commands are described in detail in Chapter 7, *SCPI Commands*.

IQ-Applications using the DDE link with the SCPI Manager will run on both the IQ-203 and a host computer with a PC Expansion card connected to an IQ-206.

The Windows DDE protocol is well documented in many books dealing with Windows programming. The basic concepts regarding DDE are similar for all programming languages. There will be, however, different syntax and commands for different languages. At EXFO, integrated applications are created using Visual Basic 4.0. The following explanations and examples are for Visual Basic.

5.2 Background Information

A DDE link for an integrated IQ application can be graphically represented as shown in Figure 5-1.

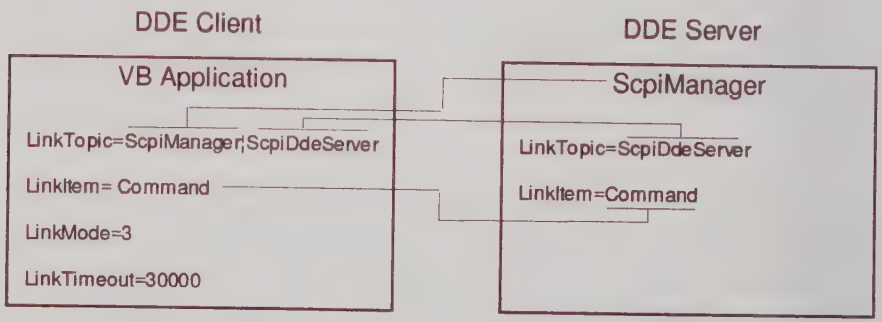

Figure 5-1. DDE Communication with SCPI Manager

As shown in the figure, there is a DDE server and a DDE client. For all integrated applications with the IQ system, the SCPI Manager is the server and the IQ-Application is the client. The type of DDE link used by the integrated application is known as a *destination link* A destination link is requested and controlled by the client application.

The following properties must be defined by the client when requesting a destination DDE link:

- link topic
- link item
- link mode
- link timeout

Link Topic

The link topic is specified in the format *application|topic*, where application is the name of the DDE server (SCPIManager) and the topic is the subject of the DDE conversation (ScpiDdeServer).

Link Item

The link item specifies the data being passed by the DDE link. For IQ applications the link item can be **Command**, **Status**, or **Lockout**. A DDE link with the item set to **Command** is used to send commands and receive responses. A DDE link with the item set to **Status** is used to read status information. **Lockout** is used to place the system in Local or Remote lockout.

***Note**: All command and status information can be linked using only link item set to "Command". See Section 5.4, Useful Programming Hints.*

Link Mode

There are three types of DDE links;

1. an automatic link (linkmode = 1), where the link data is updated every time the source value changes
2. a manual link (linkmode = 2), where data is updated only when requested
3. a notify link (linkmode = 3), where the server notifies the client that new data is available.

For IQ applications, setting linkmode = 3 offers optimum flexibility and performance.

Other Useful Functions and Events

The following functions are important to understand when programming an integrated IQ application. See the DDE programming section of your programmer's guide (depending on the programming language used) for detailed information.

- *object*.`linkexecute` used to send commands to the server
- *object*.`linkpoke` used to send data to the server
- *object*.`linkrequest` used to update a link in notify mode
- *object*_`linkopen` event occurs when a link is successfully initiated
- *object*_`linkclose` event occurs when a link has terminated
- *object*_`linkerror` event occurs when error is detected
- *object*_`linknotify` event occurs when the server has new data
- *object*_`change` event occurs when a receiving inbox value is updated

5.3 Opening a DDE Communication Channel

In order to successfully open a DDE link with the SCPI Manager;

- the IQ Optical Test System software must be open on either the IQ-203 or on a host computer equipped with the IQ-206 PC Expansion Card
- the SCPI Manager must be open with communication type set to DDE
- the appropriate link commands must be executed
- the SCPI Manager registers must be initialized

The following lines of VB code are from a simple application that opens the DDE link, initializes the SCPI Manager registers, and then polls the SCPI Manager to find out which modules are available.

*Note: The following code is from a Visual Basic form containing a textbox object called **SCPIManInBox**.*

This section establishes the link

```
Private Sub Form_Load()  'Communication with SCPI Manager

    SCPIManInBox.LinkTopic = "ScpiManager|ScpiDdeServer"
    SCPIManInBox.LinkItem = "Command"
    SCPIManInBox.LinkTimeout = 300      'timeout = 30 seconds
    SCPIManInBox.LinkMode = 3   'open DDE in NOTIFYmode (= 3)
```

This section sends commands to initialize the registers

```
    SCPIManInBox.LinkExecute "*CLS"

    SCPIManInBox.LinkExecute "*OPC"
```

Command to poll available modules

```
    SCPIManInBox.LinkExecute "INST:CAT?:FULL?"

End Sub
```

Routine to monitor the LinkNotify event and request the data

```
Private Sub SCPIManInBox_LinkNotify()

    SCPIManInBox.LinkRequest    'request data
End Sub
```

Routine to get string data in the inbox

```
Private Sub SCPIManInBox_Change()

    Dim TheInstrumentCatalog as string
    TheInstrumentCatalog= SCPIManInBox.text

End Sub
```

When this event occurs, the data requested has arrived in the SCPIManInBox

5.4 Useful Programming Hints

The following section of code illustrates three important concepts:

- integrating the module position into the SCPI command
- determining whether a command has been successfully completed
- reading error messages

```
dim SCPICommand as String
dim PMPosition as String

PMPosition="1"          'module is installed in position 1, if
                        desired this could be a variable passed by
```

```
                    an initializing routine or a application
                    launch program

SCPIManInBox.LinkExecute "*CLS"
SCPICommand="SENS"+PMPosition+":CORR:COLL:ZERO" 'Null PM
SCPIManInBox.LinkExecute SCPICommand
SCPIManInBox.LinkExecute "SYST:ERR?"
```

In the above sequence the `PMPosition` variable is used to incorporate the module position into the command. This type of command construction is important when developing applications where the physical location of a module is not predetermined.

The SYST:ERR? command will be only be executed after completion of the previous command. The string returned will be either

0, "" indicating that the command was executed successfully
or,
###, "Error message" indicating that an error was made

The following section of code describes how to put the SCPI Manager in Lockout mode.

```
SCPIManInBox.LinkTopic = "ScpiManager|ScpiDdeServer"
SCPIManInBox.LinkItem = "Lockout"
SCPIManInBox.LinkTimeout = 300      'timeout = 30 seconds

SCPIManInBox.LinkMode = 3   'open DDE in NOTIFYmode (= 3)
SCPIManInBox.text="1"
SCPIManInBox.LinkPoke
```

In order to enable lockout mode, the LinkItem is set to Lockout, and the LinkPoke command is used to enable lockout. The SCPI Manager and all the modules in communication with the SCPI Manager will be locked to their current status, either remote or local. The command:

```
SCPIManInBox.text="0"
SCPIManInBox.LinkPoke
```

will disable lockout mode.

5.5 Sample VB Program

The complete source code for a simple application can be found in the C:\IQ\APP\SAMPLE \SOURCE directory of either the IQ-203 or the host computer if the IQ Optical Test System software is installed. This application contains routines for opening the DDE link, commanding the modules, and reading data, status, and error messages.

6 THE IQ-203 AS A GPIB CONTROLLER

6.1 Introduction

As pointed out in the overview, the IQ-203 can be equipped with a GPIB controller interface card. In this configuration, the IQ-203 will operate as a fully functioning GPIB Talker/Listener/Controller while at the same time control the operation of the IQ-200 system components. Integrated applications that control both the external GPIB instruments and the IQ-200 system components can be developed and executed.

6.2 Supplied Hardware and Software

The following items are supplied as part of the IQ-203 GPIB Controller (-I2) option;

- a National Instruments AT-GPIB/TNT interface board installed in the IQ-203
- NI-488.2 software (AT-GPIB/TNT) for Windows installed on the IQ-203 hard drive
- National Instruments GPIB instruction manuals
- backup floppy disks containing the NI-488.2 software

CAUTION

The GPIB controller card must be installed at the factory. Attempting to install any other GPIB card will void the product warranty.

6.3 Configuration

The NI AT-GPIB interface card is preset for the following configuration;

- Primary address 2A0h
- Interrupt (IRQ) 11
- DMA channel 5

6.4 Programming

Integrated applications may be created in any programming environment that supports both Windows DDE linking and GPIB communication. All of the major programming languages support DDE, and commercial GPIB development kits are available for most of common programming environments.

IMPORTANT

It is important to realize that in this configuration, the IQ modules are not GPIB instruments. Integrated control of the modules is accomplished through a DDE link as described in Chapter 5, *Integrated Applications (DDE).*

7 SCPI COMMANDS

7.1 Standard Status Data Structure

Figure 7-1 on the following page illustrates the four common status and enable registers as defined by IEEE 488.2. This diagram is a useful aid in understanding the common commands and how a service request (SRQ) is generated. The four registers are;

1. Standard Event Status Register (ESR)
2. Standard Event Status Enable Register (ESE)
3. Status Byte Register (STB)
4. Service Request Enable Register (SRE)

BIT	ESR	ESE	STB	SRE
0	Operation Complete	Operation Complete	Source Status	Source Status
1	Request Control	Request Control		
2	Query Error	Query Error		
3	Device Dependent Error	Device Dependent Error	Questionable Status	Questionable Status
4	Execution error	Execution error	Event Status Bit	Event Status Bit
5	Command Error	Command Error	Message Available	Message Available
6	User Request	User Request		Request Service/Master Summary Status
7	Power On	Power On	Operation Status	Operation Status

Table 7-1. Standard Registers

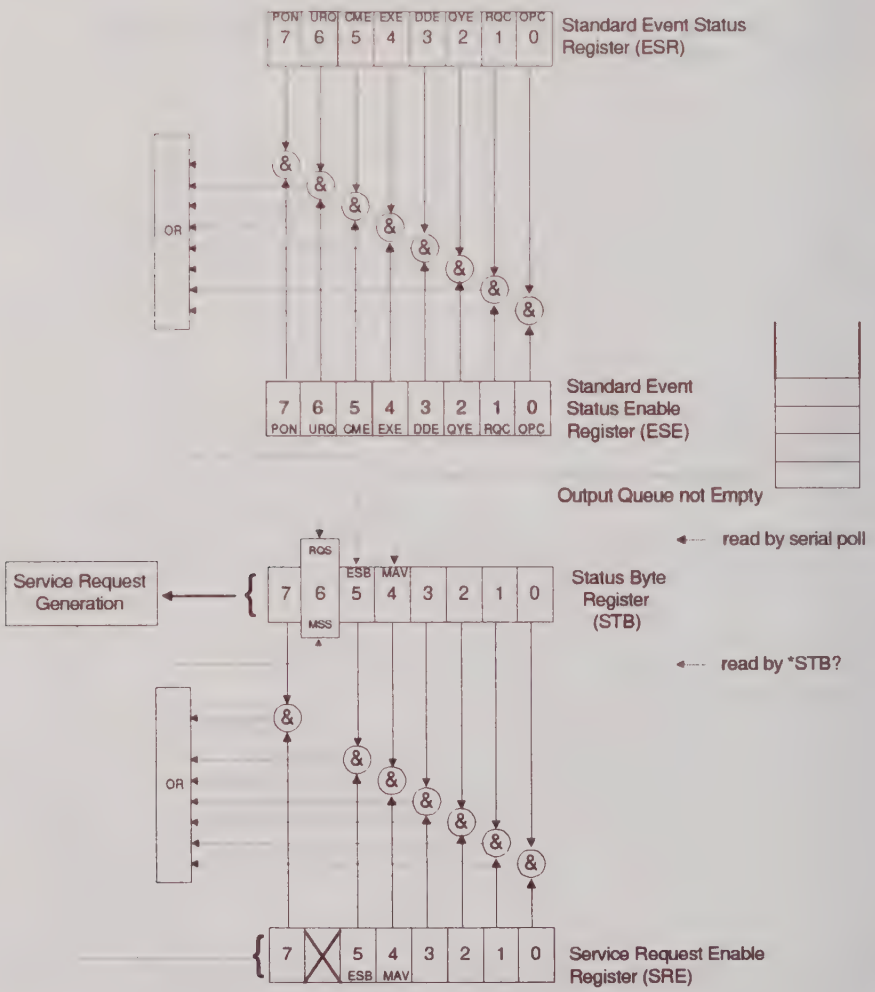
PON URQ CME EXE DDE QYE RQC OPC
7 6 5 4 3 2 1 0
Standard Event Status Register (ESR)
OR
7 6 5 4 3 2 1 0
PON URQ CME EXE DDE QYE RQC OPC
Standard Event Status Enable Register (ESE)
Output Queue not Empty
read by serial poll
Service Request Generation
RQS
ESB MAV
7 6 5 4 3 2 1 0
MSS
Status Byte Register (STB)
read by *STB?
OR
7 5 4 3 2 1 0
ESB MAV
Service Request Enable Register (SRE)

Figure 7-1. Standard Status Data Structures (IEEE 488.2)

An SRQ is forced when a bit is set in the STB and at the same time the corresponding SRE bit is set. when the SRQ is generated, the RQS bit is set to 1, and remains set until read by a serial poll. Once the RQS is read, it returns to 0.

7.2 Common Commands

The SCPI Manager recognizes all of the common commands identified as mandatory by IEEE 488.2. These commands, as well as some optional common commands, are summarized in Table 7-2.

COMMAND	FUNCTION
*CLS	Clear Status Command
*ESE	Standard Event Status Enable Command
*ESE?	Standard Event Status Enable Query
*ESR?	Standard Event Status Register Query
*IDN?	Identification Query
*OPC	Operation Complete Command
*OPC?	Operation Complete Query
*RST	Reset Command
*SRE	Service Request Enable Command
*SRE?	Service Request Enable Query
*STB?	Read Status Byte Query
*TRG	Trigger Command
*TST?	Self Test Query
*WAI	Wait to Continue Command

Table 7-2. Common Command Summary

The common commands are fully explained on the following pages.

*CLS

Description This command sets the contents of the Standard Event Register (ESR), the Status Byte Register (STB), and the Error Queue (ERR) to zero. This command is commonly used to clear the status registers before enabling SRQ. Note that the output queue, the Standard Event Status Enable Register (ESE), and the Service Request Enable Register (SRE) are not affected

Syntax *CLS

*ESE

Description This command is used to set bits in the Standard Events Status Enable Register (ESE) to a new value. The contents of the ESE register are logically ANDed with the ESR register. A non zero result will set the Event Summary Bit (ESB) of the Status Byte register. This command is useful for selecting which events may generate an SRQ.

Syntax *ESE<space><value>

Parameter The <value> parameter must be between 0 and 255

*ESE?

Description This query reads the contents of the Events Status Register.

Syntax *ESE

Response Binary integer with a value between 0 and 255.

*ESR?

Description This query reads the contents of the Standard Events Register.

Syntax *ESR

Response Binary integer with a value between 0 and 255.

*IDN?

Description This query reads the IQ system identification string.

Syntax *IDN

Response "EXFO E.O. Eng IQ-203 Vx.x" where x.x is the current product version

*OPC

Description This command will enable the SCPI Manager to respond to the *OPC? query.

Syntax *OPC

*OPC?

Description This query sets the operation complete bit in the ESR (bit 0) and puts an ASCII 1 in the output queue, when the contents of the input queue have been processed. This command is useful to prevent another command from processing until the current command is complete.

Syntax *OPC?

Response "1"

*RST

Description This command empties the step responses list in the SCPI Manager. This command is only seen when it is a part of another multiple command. In the example below, by adding this command after DATE?, you will not be able to access the answer. The *RST, in this instance erases the date. In addition, this command also performs the following operations:

1. Return to initial state before command was sent, and not necessarily to previous settings
2. Force the device to enter into an Operation Complete Command Idle State (OCIS)
3. Force the device to enter into an Operation Complete Query Active State (OQIS)
4. Initialize previous responses unless there has been a program message terminator preceded by an *RST.

Syntax *RST

Example SHEL:DATE?;*RST <NL>

*SRE

Description This command sets bits in the Service Request Enable register and enables the corresponding bit in the Status Register. The command can be used to select which events can initiate a service request.

Syntax *SRE<space><value>

Parameter The <value> parameter must be between 0 and 255

*SRE?

Description This query returns the contents of the Service Request Enable register.

Syntax *SRE?

Response A binary integer between 0 and 255

*STB?

Description This query returns the contents of the Status Byte register.

Syntax *STB?

Response A binary integer between 0 and 255

*TRG

Description This command will trigger any event that was suspended and is waiting for a trigger.

Syntax *TRG

Parameters

*TST?

Description This query initiates an internal self test, and returns a binary value indicating the results of the test.

Syntax *TST?

Response A binary value;
"0" test complete with no errors
"1" test complete with errors

*WAI

Description This command prevents the SCPI Manager from processing any further commands until the No-Operation-Pending-Flag is set. All pending operations must be complete before processing another command.

Syntax *WAI

Parameters None

7.3 System Commands

SYST:ERR?

Description This query returns the next error contained in the FIFO error message buffer.

Syntax SYST:ERR?

Response An error message as described in Chapter 8, *Error Messages*,.

SYST:VERS?

Description This query returns the current system software version.

Syntax SYST:VERS

Response A string identifying the system software version.

7.4 IQ-200 SCPI Command Structure

The IQ-200 commands follow the guidelines determined by the Standard-Commands-for-Programmable-Interface (SCPI) consortium. Due to the modular nature of the IQ-200 product family, the command format is, however, slightly modified so that specific modules in the system may be addressed.

The IQ-203, connected to 4 IQ-206 expansion units, may accommodate a maximum of 27 modules. These modules may be the same type (e.g. 27 source modules) or they may be different modules. The SCPI commands for the IQ modules incorporate a module address modifierat the end of the command subsystem identifier to specify a particular module. For example the command syntax,

FORM(0..26):READ[:DATA]<space><digits>

is used to change the measurement display resolution (number of digits after the decimal point) for an IQ-1100 Power Meter.

In this particular example;

- FORM identifies that the command is a part of the SCPI FORMat subset of commands
- (0..26) is the module address modifier and identifies the specific module to which the command is addressed

***Note**: Commands sent to the IQ Optical Test System do not require the address modifier.*

- READ and DATA are keywords that define the function of the command
- [ ] indicates that a keyword or a parameter is optional
- <space> is included to indicate that a space is required

- <digits> is the command parameter
- Keywords must be separated by a colon
- All keywords must be CAPITAL characters

A typical command would be;

FORM2:READ:DATA 1

This command instructs the module located in IQ position 2, to display a power measurement with 1 digit after the decimal point.

7.4.1 Module SCPI Command Address

As previously mentioned, the IQ-200 system can accommodate up to 27 modules. This maximum configuration is shown in Figure 7-2. The physical location and the module SCPI command address of each module is shown in the diagram.

***Note**: When using the IQ-206 Expansion Unit and the IQ-200 PC Expansion Card connected to a , modules 0, 1, and 2 do not exist. The module located in slot A of the first IQ-206 (1-A) is addressed as module 3 with the rest of the modules following in sequence. The 0,1, and 2 addresses are reserved for modules installed in the IQ-203 mainframe.*

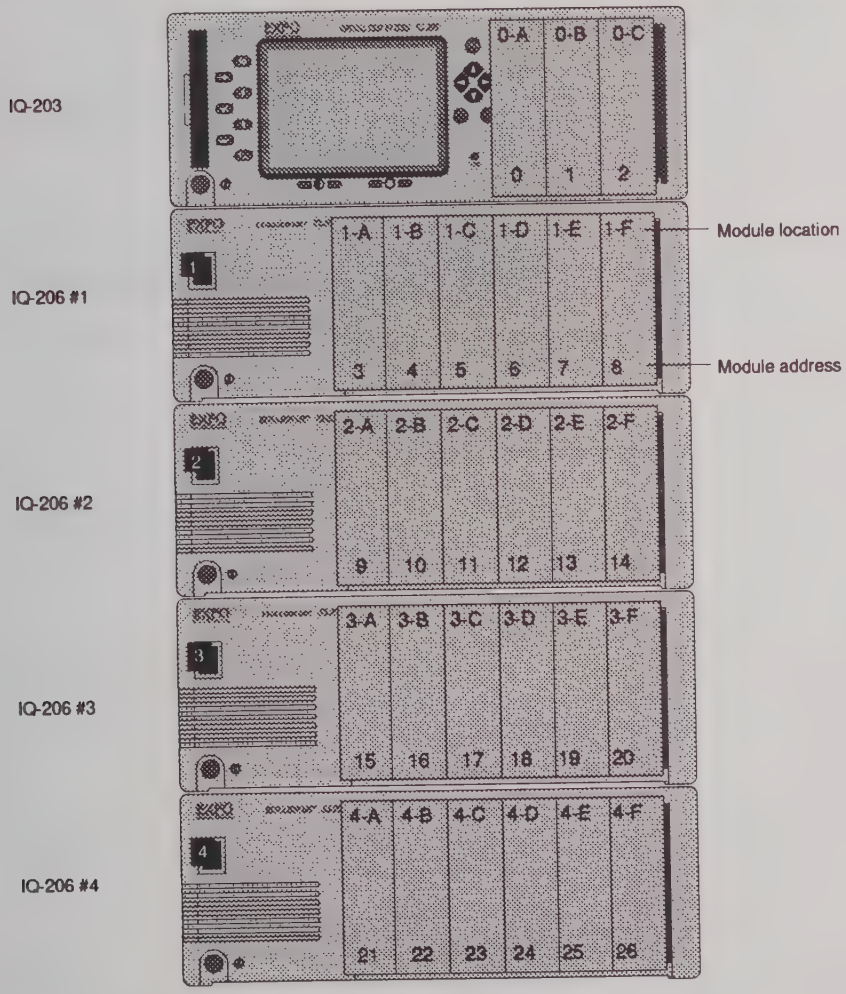

Figure 7-2. IQ-200 Module Addresses

7.5 IQ-200 Optical Test System Commands

The following commands are addressed to the IQ operating system otherwise known as the IQ Optical Test System. It should be noted that these commands do not require the command address modifier.

INSTrument:CATalog?

Description This query returns a comma separated list of strings identifying the available modules.

Syntax INST:CAT?

Response "Module_type, ..." identifies both the module part number and the module physical location.

Example INST:CAT?
"IQ-2123-BLC (0-A)"
-this response indicates that a dual wavelength Fabry-Perot source is located in the IQ-203 slot A (see Figure 7-2)

INSTrument:CATalog:FULL

Description This query returns a list of available modules along with each module logical instrument number.

Syntax INST:CAT:FULL?

Response "module_number - module_type, ..." where module_number refers to the logical instrument number and the module_type identifies both the module part number and the module physical location.

Example INST:CAT:FULL?
"0 - IQ-2123-BLC (0-A)"
-this response indicates that a dual wavelength Fabry-Perot source is located in the IQ-203 slot A and that its logical instrument number is 0 (see Figure 7-2)

INSTrument:STATe

Description This command opens or closes the software application that is required to operate a specific module.

Syntax INST:STAT <space><boolean>,<address>

Parameters The <boolean> parameter is:
"1" or "ON" opens the application
"0" or "OFF" - closes the application

The <address> parameter is a number (0..26) corresponding to the module logical number.

Example INST:STAT 1,1
-activates the application for the logical module 1 (IQ-203 slot B) (see Figure 7-2)

INSTrument:STATe?

Description This query returns a value indicating whether the application for a specific module is open or closed.

Syntax INST:STAT?<space><address>

Parameters The <address> parameter is a number (0..26) corresponding to the module location.

Response
"1" - the application is open
"0" - the application is closed

Example INST:STAT? 1

SHELl:DATE

Description This command sets the IQ system date.

Syntax SHEL:DATE<space><YYYY>,<MM>,<DD>

Parameters
YYYY - the year
MM - the month (01-12)
DD - the day (01-31)

Example SHEL:DATE 1996,02,11

SHEL:DATE?

Description This query returns the IQ system date.

Syntax SHEL:DATE?

Response "MM/DD/YY" where
MM - the month (01-12)
DD - the day (01-31)
YY - the year

Example SHEL:DATE?
"01/12/96" for 12 February, 1996

SHEL:LIGHt

Description This command turns off or on the IQ-203/IQ-206 front panel indicating lights, including the integrated VGA display and the module LED push buttons. The external monitor and keyboard are not affected by this command.

Syntax SHEL:LIGH<space><boolean>

Parameters The <boolean> parameter can be:
1 or ON - turns on front panel lights
0 or OFF - turns off front panel lights

Example SHEL:LIGH OFF

SHELl:LIGHt?

Description This query returns the state of the front panel indicating lights.

Syntax SHEL:LIGH?

Response "1" - front panel lights are on
"0" - front panel lights are off

SHELl:TIME

Description This command sets the system time using the 24 hour clock.

Syntax SHEL:TIME<space><HH>,<MM>,<SS>

Parameters

HH	- the hours (00 - 23)
MM	- the minutes (00 - 59)
SS	- the seconds (00 - 59)

Example SHEL:TIME 14,23,00

SHEL:TIME?

Description This query returns the IQ system time.

Syntax SHEL:TIME?

Response "HH:MM:SS" where
HH - the hours
MM - the minutes
SS - the seconds

Example SHEL:TIME?

7.6 IQ Power Meter Module Commands

These commands are applicable to all IQ power meters, including the IQ-1100 single channel, IQ-1200 4 Channel, and the IQ-1500 Calibration Power Meter modules.

ABORt

Description This command stops any measurement in process. For the IQ-1200 4-Channel Power Meter, this command stops the measurement on all 4 channels.

Syntax ABORt(0..26)

Parameters N/A

Example ABOR1

See also INIT:CONT; INIT:CONT?; FETC:SCAL:POW:DC?

FETCh:[SCAL]ar:POWer: DC?

Description This query returns the stored value (for the selected channel when using the IQ-1200 4-Channel Power Meter).

Syntax FETC(0..26):[SCAL:]POW: DC?

Response A power measurement in the exponential format; "±999.999E±09" in the selected units. The number of digits after the decimal point depends on the selected resolution.

Example FETC1:SCAL:POW: DC?

See also INIT:CONT; INIT:CONT?; INIT:IMM

FORMat:READings[:DATA]

Description This command changes the resolution of the displayed power value, when units of dB or dBm are selected (for the selected channel when using the IQ-1200 4-Channel Power Meter).

Syntax FORM(0..26):READ[:DATA]<space><digits>

Parameters: The <digits> parameter can be:
0 -zero digits after the decimal point
1 -one digit after the decimal point
2 -two digits after the decimal point
3 -three digits after the decimal point
4 -auto resolution, determined by the measured power level

Example FORM1:READ:DATA 3

INITiate:CONTinuous

Description This command starts or stops continuous power measurements (for all channels if using the IQ-1200 4-Channel Power Meter).

Syntax INIT(0..26):CONT<space><boolean>

Parameters The <boolean> value refers to;
0- stop
1- start

Example INIT1:CONT 1

See also FETC:SCAL:POW: DC?; INIT:CONT?; INIT:IMM

INITiate:CONTinuous?

Description This query returns a value indicating whether continuous power measurements are in progress (for all channels if using the IQ-1200 4-Channel Power Meter).

Syntax INIT(0..26):CONT?

Response 0- stopped
1- started

Example INIT1:CONT?

See also FETC:SCAL:POW: DC?; INIT:CONT; INIT:IMM

INITiate:[IMMediate]

Description This command stores a power measurement value (for all channels if using the IQ-1200 4-Channel Power Meter).

Syntax INIT(0..26):[IMM]

Parameters N/A

Example INIT1:IMM

See also FETC:SCAL:POW: DC?; INIT:CONT

INST:NSELect

Description This command selects a power meter channel when using the IQ-1200 4-channel Power Meter. Until a new channel is selected, any subsequent commands will be directed to this channel.
This command is specific to the IQ-1200 4-Channel Power Meter. An error message appears if the command is sent to a single channel power meter.

Syntax INST(0..26):NSEL<space><numeric_value>

Parameters The <numeric_value> can be 1,2,3,or 4.

Example INST1:NSEL 4

MMEMory:ACQuisition

Description This command initiates a data acquisition. Acquisition data, at the selected sampling rate, will be recorded to the system hard disk in the file **C:\IQ\USERFILE\GPIBPM##.DAT**, where ## is the logical module number (0..26). The acquisition will continue for 99 hours, 59 minutes, and 59 seconds or until a acquisition stop (MMEM:ACQ 0) command is transmitted.

Syntax MMEM(0..26):ACQ<space><boolean>

Parameters The <boolean> parameter can be:

1 -start the data acquisition

0 -stop the data acquisition

Example MMEM0:ACQ 1

MMEMory:ACQuisition?

Description This query returns a value indicating whether data acquisition is in progress.

Syntax MMEM(0..26):ACQ?

Response The response is in the format;

"1" - data acquisition is in progress

"0" - data acquisition is not in progress

Example MMEM0:ACQ?

READ:[SCAL]ar:POWer: DC?

Description This command performs an initiate and fetch (for the selected channel when using the IQ-1200 4-Channel Power Meter). A measurement value is stored and returned.

Syntax READ(0..26):[SCAL:]POW: DC?

Response A power measurement in the exponential format; "±999.999E±09" in the selected units. The number of digits after the decimal point depends on the selected resolution.

Example FETC1:SCAL:POW: DC?

SENSe:AVERage[:STATe]

Description This command activates or deactivates data averaging. It is specific to the IQ-1100 single channel power meter.

Syntax SENS(0..26):AVER[:STAT]<space><boolean>

Parameters The < boolean > parameter can be:
0 -averaging is disabled
1 -averaging is enabled

Example SENS1:AVER:STAT 1

SENSe:AVERage:STATe?

Description This query returns a value indicating whether data averaging is enabled or disabled. It is specific to the IQ-1100 single channel power meter.

Syntax SENS(0..26):AVER:STAT?

Response 0 -averaging is disabled
1 -averaging is enabled

Example SENS1:AVER:STAT?

SENSe:CORRection:COLLect:ZERO

Description This command performs an OFFSET NULL measurement (for the selected channel when using the IQ-1200 4-Channel Power Meter). If light is detected, the error message, "Light on detector" appears.

Syntax SENS(0..26):CORR:COLL:ZERO[ALL]

Parameters The optional [ALL] parameter will perform the NULL on all four channels of the IQ-1200.

Example SENS1:CORR:COLL:ZERO

SENSe:CORRection:OFFSet[:MAGNitude]

Description This command sets an offset value that is applied when units of dB or dBm are selected (for the selected channel when using the IQ-1200 4-Channel Power Meter).

Syntax

SENS(0..26):CORR:OFFS[:MAGN]
<space><numeric_value>
[<space>DB]

Parameters The <numeric_value> is an offset with units of dB. -99.999 ≤ numeric_value ≤ 99.999 (dB). If an invalid parameter is entered, the error message, "Data Type Error" appears.

Example

SENS1:CORR:OFFS:MAGN 22.105

SENSe:FREQuency

Description This command selects a data sampling rate. It is specific to the IQ-1100 single channel power meter.

Syntax SENS(0..26):FREQ<space><numeric_value>[<space>Hz]

Parameters The <numeric_value> is the sampling rate with units of sec^{-1}. 0.1, 0.5, 1, 2, 5, 10, 20, 40 samples per second. If an invalid parameter is entered, the error message, "Data Type Error" appears.

Example SENS1:FREQ 20

SENSe:FREQuency:CATAlog?

Description This query returns a list of available sampling rates.

Syntax SENS(0..26):FREQ:CATA?

Response List of available sampling rates in the format; "40.0;20.0;10.0;5.0;1.0;0.5;0.1"

Example SENS1:FREQ:CATA?

SENSe:FREQuency?

Description This query returns the current sampling rate.

Syntax SENS(0..26):FREQ?

Response Returns the current sampling rate in units of samples/second

Example SENS1:FREQ?
"20.0"

SENSe:POWer:REFerenc:DISPlay

Description This command performs a new reference measurement for the current wavelength and changes the display to read relative power (units of dB) (for the selected channel when using the IQ-1200 4-Channel Power Meter).

Syntax SENS(0..26):POW:REF:DISP

Parameters N/A

Example SENS1:POW:REF:DISP

SENSe:POWer:REFerenc:STATe

Description This command selects whether absolute (dBm) or relative power measurements are performed (for the selected channel when using the IQ-1200 4-Channel Power Meter).

Syntax SENS(0..26):POW:REF:STAT<space><boolean>

Parameters The <boolean> value represents either dB or dBm
0- absolute or dBm
1- relative or dB

Example SENS1:POW:REF:STAT

SENSe:POWer:REFerenc:STATe?

Description This query returns a value indicating if the power meter is displaying absolute (dBm) or relative power values (for the selected channel when using the IQ-1200 4-Channel Power Meter).

Syntax SENS(0..26):POW:REF:STAT?

Response "0" absolute (dBm or watt)
"1" relative (dB)

Example SENS1:POW:REF:STAT?

SENSe:POWer:REFerence?

Description This query returns the reference power for the current wavelength (for the selected channel when using the IQ-1200 4-Channel Power Meter).

Syntax SENS(0..26):POW:REF?

Response The current reference value in dBm which can be any value within the power range of the power meter in the format; "99.999 dBm"

Example SENS1:POW:REF?
"49.428 dBm"

SENSe:POWer:UNIT

Description This command changes the absolute measurement display units dBm or watt (for the selected channel when using the IQ-1200 4-Channel Power Meter).

Syntax SENS(0..26):POW:UNIT<space><value>

Parameters The <value> can be

DBM or 0	- sets power display to dBm
W or 1	- sets power display to watts (pw, nw, μw, or mw)

Example SENS1: POW:UNIT DBM

SENSe:POWer:UNIT?

Description This query returns the current absolute power measurement display units (for the selected channel when using the IQ-1200 4-Channel Power Meter).

Syntax SENS(0..26):POW:UNIT?

Response "0" - current units are dBm
"1" - current units are watts (pw, nw, µw, or mw)

Example SENS1: POW:UNIT?

SENSe:POWer:WAVElength

Description This command selects a new operating wavelength (for the selected channel when using the IQ-1200 4-Channel Power Meter).

Syntax SENS(0..26):POW:WAVE<space><numeric_value> [<space>NM]

Parameters The <numeric_value> is an operating wavelength with units of nm`s. Any wavelength within spectral range of the power meter optical detector at 1 nm resolution (0.1 nm for the IQ-1500 Calibration Power Meter) may be selected. See the power meter instruction manual for the exact spectral range for each detector type. An invalid parameter will cause a "Data Type Error".

Example SENS1: POW: WAVE 1310

SENSe:POWer:WAVElength?

Description This query returns the currently selected calibrated wavelength (for the selected channel when using the IQ-1200 4-Channel Power Meter).

Syntax SENS(0..26):POW:WAVE?

Response The current wavelength in nanometers (nm) in the format; "9999 nm" (9999.9 for the IQ-1500)

Example SENS1:POW:WAVE?
"1310 nm"

UNIT:POWer

Description This command changes the measurement display units (for the selected channel when using the IQ-1200 4-Channel Power Meter).

Syntax UNIT(0..26):POW<space><units>

Parameters The <units> parameter can be:
W - measured value displayed in watts (pw, nw, µw, or mw)
DB - measured value displayed in dB relative to the current reference
DBM- measured value displayed in dBm

Example UNIT1:POW DBM

7.7 IQ-3200 Return Loss Meter Module Commands

ABORt

Description This command stops any measurement currently in process.

Syntax ABORt(0..26)

Parameters N/A

Example ABOR1

FETCh?

Description This query returns a stored optical return loss value.

Syntax FETC(0..26)?

Response ORL measurement value in the format;
"999.999"

Example FETC1?

FORMat:READings:DATA

Description This command changes the resolution of the Optical Return Loss measurement.

Syntax FORM(0..26):READ:DATA<space><digits>

Parameters The <digits> parameter can be:
0 -zero digits after the decimal point
1 -one digit after the decimal point
2 -two digits after the decimal point
3 -three digits after the decimal point
4 -autoresolution, determined by the measured value

Example FORM1:READ:DATA 3

FORMat:READings:DATA?

Description This query returns the current measurement resolution.

Syntax FORM(0..26):READ:DATA?

Response "0" -zero digits after the decimal point
"1" -one digit after the decimal point
"2" -two digits after the decimal point
"3" -three digits after the decimal point
"4" -auto resolution, determined by the measured power level

Example FORM1:READ:DATA?

INITiate:CONTinuous

Description This command starts or stops continuous ORL measurements.

Syntax INIT(0..26):CONT<space><boolean

Parameters The <boolean> value refers to;
0 or OFF - stop measurements
1or ON - start measurements

Example INIT1:CONT 1

INITiate:CONTinuous?

Description This query returns a value indicating whether continuous ORL measurements are in progress.

Syntax INIT(0..26):CONT?

Response "0"- continuous measurements not in progress
"1"- continuous measurements are in progress

Example INIT1:CONT?

INITiate[:IMMediate]

Description This command stores a single return loss measurement value. Note that the IMM command is optional.

Syntax INIT[:IMM]

Parameters N/A

Example INIT1:IMM

INPut:OFFSet

Description This command selects an offset value that will be added to (or subtracted from) the ORL measurement.

Syntax INP(0..26):OFFS<space><numeric_value>[<space>DB]

Parameters The <numeric_value> parameter is the desired offset in the format "±999.999". The units [DB] are optional.

Example INP1:OFFS 6.2 DB

READ:POWer?

Description This command stores and fetches (performs an INIT:IMM and a FETC?) an ORL measurement value.

Syntax READ(0..26): POW?

Response ORL measurement value (999.999)

Example READ1:POW?

SENSe:AVERage:STATe

Description This command activates or deactivates data averaging.

Syntax SENS(0..26):AVER:STAT <space><boolean>

Parameters The < boolean > parameter can be:
0 -averaging is OFF
1 -averaging is ON

Example SENS1:AVER:STAT 1

SENSe:AVERage:STATe?

Description This query returns a value indicating whether data averaging is on or off.

Syntax SENS(0..26):AVER:STAT?

Response "0" -averaging is OFF
"1" -averaging is ON

Example SENS1:AVER:STAT?

SENSe:CORRection:COLLect:ZERO

Description This command performs an OFFSET NULL measurement.

Syntax SENS(0..26):CORR:COLL:ZERO
Parameters N/A

Example SENS1:CORR:COLL:ZERO

SENSe:POWer:REFerence

Description This command performs the IQ-3200 reference calibration. The command format includes parameters that also sets the calibration reference reflection value. When calibrating with the EXFO supplied reference reflection jumper, the module software automatically determines the correct calibration value based on the currently selected wavelength.

Syntax SENS(0..26):POW:REF<space><value>[<space><units>]

Parameters The <value> parameter can be,
"EXFO" - use the supplied reference reflection. The numeric value will be provided by the module software.

or

"±99.999"- a numeric value expressing the reference reflection. When a numeric value is entered, the <units> parameter must be specified;
"DB" - the reference reflection is expressed in dB
"%" - the reference reflection is expressed in %

Example SENS1:POW:REF EXFO

SENSe:POWer:REFLection-free

Description This command performs a zero measurement, eliminating the effects of parasitic reflections. A zero measurement can be performed for each wavelength (maximum of 2).

Syntax SENS(0..26):POW:REFL

Parameters N/A

Example SENS1:POW:REFL

SENSe:POWer:WAVElength:LIST?

Description This query returns the list of programmed wavelengths. The maximum number of values in the list is 2. Note that executing this command does not activate the wavelength.

Syntax SENS(0..26):POW:WAVE:LIST?

Response There are two wavelengths in the format:
"9999 - 9999" - the first number is wavelength 1 and the second value is wavelength 2.

Example SENS1:POW:WAVE:LIST?

SENSe:POWer:WAVElength:PROGram

Description This command programs the two available wavelengths. Any two wavelengths between 1250 nm and 1600 nm at 1 nm resolution can be selected

Syntax SENS(0..26):POW:WAVE:PROG<space><sel_lam>,<value> [<space>NM]

Parameters The < sel_lam > parameter identifies which of the programmed wavelengths will be modified. Acceptable values are **1** or **2**.

The < value> is a calibrated wavelength. Any wavelength within the spectral range of the IQ-3200 (1250-1600 nm) at 1 nm resolution can be used. The units [NM] are optional.

Example SENS1:POW:WAVE:PROG 1,1310 NM
This commands sets wavelength number 1 to 1310 nm.

SENSe:POWer:WAVElength:SET

Description This command selects a new wavelength. The wavelength must be one of the two programmed values.

Syntax SENS(0..26):POW:WAVE:SET<space><numeric_value>[<space>NM]

Parameters The <numeric_value> is a calibrated wavelength and must be one of the two programmed values. The units [NM] are optional.

Example SENS1:POW:WAVE:SET 1310 NM

SENSe:POWer:WAVElength?

Description This command returns the currently selected wavelength.

Syntax SENS(0..26):POW:WAVE?

Response The current wavelength in nanometers (nm) in the format: "9999"

Example SENS1:POW:WAVE?

7.8 IQ-2100 Source Commands

SOURce:AM[:INTernal]:FREQuency

Description This command selects the internal modulation frequency. The internal modulation is 50 % duty cycle at the selected frequency.

Syntax SOUR(0..26):AM[:INT]:FREQ<space><value> [<space><units>]

Parameters The available modulation frequencies are:
270Hz, 1000Hz, 2000Hz, and CW (for no modulation)

Allowable [units] are Hz or kHz

Example SOUR1:AM:INT:FREQ 270 Hz (set modulation to 270 Hz) or
SOUR1:AM:INT:FREQ CW (set modulation to none)

SOURce:AM[:INTernal]:FREQuency?

Description This query returns a value indicating the present internal modulation frequency.

Syntax SOUR(0..26):AM[:INT]:FREQ?

Response A four digit integer identifying the frequency in Hz:
"9999"

Example SOUR1:AM:FREQ?

SOURce:POWer:ATTenuation

Description This command changes the source internal attenuation. The source power is at its maximum when the attenuation is set to 0.0 dB.

Syntax SOUR(0..26):POW:ATT<space><value>[<space><units>]

Parameters The <value> parameter is between 0.0 and 10.0 for Laser sources and between 0.0 and 5.0 for LED sources. The attenuation is a positive value.
The [units] parameter is DB and optional

Example SOUR1:POW:ATT 5.2 DB

SOURce:POWer:ATTenuation?

Description This query returns the internal source power attenuation value.

Syntax SOUR(0..26):POW:ATT?

Response A dB value between 0.0 and 10.0 (units of dB) in the format: "99.9"

Example SOUR1:POW:ATT?

SOURce:POWer:STATe

Description This command turns ON or OFF the optical source. When the source is on, the red LED (Active) on the front of the module illuminates.

Syntax SOUR(0..26):POW:STAT<space><boolean>

Parameters The <boolean> value can be;
"0" or "OFF" - turn off the source
"1" or "ON" - turn on the source

Example SOUR1:POW:STAT ON

SOURce:POWer:STATe?

Description This query returns a value indicating the status of the optical source, ON or OFF

Syntax SOUR(0..26):POW:STAT?

Response "0" - the source is OFF
"1" - the source is ON

Example SOUR1:POW:STAT?

SOURce:POWer:WAVElength

Description This command selects a wavelength when using a dual wavelength source module (IQ-2112 or IQ-2123).

Syntax SOUR(0..26):POW:WAVE<space><value>

Parameters The <value> parameter can be;
"UPP" - switch to the larger wavelength
"LOW" - switch to the smaller wavelength

Example SOUR1:POW:WAVE UPP

SOURce:POWer:WAVElength?

Description This query returns a value indicating which wavelength is currently selected.

Syntax SOUR(0..26):POW:WAVE

Response The response is a four digit number identifying the current wavelength (units are nm) in the format:
"9999"

Example SOUR1:POW:WAVE?

7.9 IQ-3100/3300 Variable Attenuator/Reflector Commands

The IQ-3100 Variable Attenuator and IQ-3300 Variable Reflector are modules that share the same commands. The parameters in the following commands can be either attenuation values (IQ-3100) or reflection values (IQ-3300), depending on the module type.

IMPORTANT

For the purpose of these commands, both attenuation and reflection (return loss) are considered to be positive values.

INPut:ARESolution?

Description This query returns the smallest attenuation or reflection step available. Use this command to determine the maximum resolution of the attenuation or reflection setting.

Syntax INP(0..26):ARES?

Response An attenuation or reflection value in the format: "9.999". The units are dB.

Example INP1:ARES?

INPut:ATTenuation

Description This command sets the attenuation or reflection to a specific value. The valid range of values depends on the type of module, the configuration, and the current wavelength. See the appropriate module instruction manual. The resolution of the value is 0.005 dB for singlemode attenuators, and 0.01 dB for multimode attenuators and variable reflectors.

Syntax INP(0..26):ATT<space><value>[<space><units>]

Parameters The <value> parameter can be;
-a valid attenuation or reflection value followed by the units [DB]
or
MIN -sets the module to the minimum setting for the current wavelength
MAX -sets the module to the maximum setting for the current wavelength

Example INP1:ATT -15.355 DB

INPut:ATTenuation?

Description This query returns a value indicating the current attenuation or reflection setting OR the minimum/maximum attenuation or reflection setting.

Syntax INP(0..26):ATT? [<space><value>]

Parameters The <value> parameter is optional and can be;
MIN -obtains the minimum setting for the current wavelength
MAX -obtains the maximum setting for the current wavelength

Response The attenuation or reflection value in the format: "999.999". The units are in dB.

Example INP1:ATT? MIN or
INP1:ATT?

INPut:OFFSet

Description This command sets a reference value. When the module is set to relative mode, the displayed attenuation or reflection is relative to this reference value.

Syntax INP(0..26):OFFS<space><value>[<space><DB>]

Parameters The <value> parameter can be:
a valid attenuation or reflection value followed by the (optional) units [DB].

Example INP1:OFFS 12.345 DB

INPut:OFFSet:DISPlay

Description This command sets the reference value to the current absolute attenuation or reflection setting. The displayed attenuation or reflection will be relative to the reference.

Syntax INP(0..26):OFFS:DISP

Parameters N/A

Example INP1:OFFS:DISP

INPut:OFFSet?

Description This query returns the current reference value.

Syntax INP(0..26):OFFS?

Response The current reference in the format: "999.999". The units are dB.

Example INP1:OFFS?

INPut:WAVElength

Description This command selects a specific calibrated wavelength. The wavelength range depends on the type and configuration of the module. See the appropriate module Instruction Manual for further information.

Syntax INP(0..26):WAV<space><numeric_value>[<space><NM>]

Parameters The <numeric_value> parameter is the actual wavelength in the format 9999.9[NM]. The units are optional.

Example INP1:WAV 1310 NM

INPut:WAVElength?

Description This query returns a value indicating the currently selected wavelength, the minimum available wavelength, or the maximum available wavelength.

Syntax INP(0..26):WAV?[<space><min_or_max>]

Parameters The <min_or_max> parameter is optional and can be;
"MIN" - requests the minimum wavelength
"MAX" - requests the maximum wavelength

Response The wavelength in the format:
"9999.9". The units are nm.

Example INP1: WAV? MAX or
INP1: WAV?

OUTPut:APMode

Description This command selects absolute or relative attenuation or reflection display. When absolute is selected the absolute attenuation (or reflection) introduced by the module is displayed. When relative display is selected, attenuation (or reflection) relative to the current reference is displayed.

Syntax OUTP(0..26):APM<space><boolean>

Parameters The <boolean> parameter can be:
"0" or "OFF" - relative display mode
"1" or "ON" - absolute display mode

Example OUTP1:APM ON

OUTPut:APMode?

Description This query returns a value indicating whether or not absolute display mode is selected.

Syntax OUTP(0..26):APM?

Response "0" - relative display mode
"1" - absolute display mode

Example OUTP1:APM?

OUTPut[:STATe]

Description This command controls the shutter. With an attenuator, the shutter blocks optical continuity. For a Variable Reflector, the shutter removes the reflection generating mirror from the circuit and optical continuity is maintained.

Syntax OUTP(0..26)[:STAT]<space><boolean>

Parameters The <boolean> parameter can be:
"0" or "OFF" - the shutter is optically closed
"1" or "ON" - the shutter is optically open

Example OUTP1:STAT ON

OUTPut[:STATe]?

Description This query returns the state of the instrument shutter.

Syntax OUTP(0..26)[:STAT]?

Response "0" - the shutter is optically closed
"1" - the shutter is optically open

Example OUTP1:STAT?

PROGram:SELected:EXECute

Description This command starts execution of the currently loaded attenuation (or reflection) program.

Syntax PROG(0..26):SEL:EXEC

Parameters N/A

Example PROG1:SEL:EXEC

PROGram:SELected:NAME

Description This command loads a previously defined attenuation (or reflection) program into the module's internal memory.

Syntax PROG(0..26):SEL:NAME<space><string>

Parameters The <string> parameter is the path and filename of the program to be loaded

Example PROG1:SEL:NAME C:\IQ\USERFILE\PROGFVA.PRG

PROGram:SELected:NAME?

Description This query returns the name and path of the currently loaded attenuation or reflection program.

Syntax PROG(0..26):SEL:NAME?

Response Returns a string identifying the path and name of the program file in the format
"C:\IQ\USERFILE\PROGFVA.PRG"

Example PROG1:SEL:NAME?

PROGram:SELected:STATe

Description This command controls the status of the currently loaded program.

Syntax PROG(0..26):SEL:STAT<space><value>

Parameters The <value> parameter can be:
"0" or "RUN" - start the program
"1" or "PAUSe" - pause the program
"2" or "STOP" - stop program execution
"3" or "CONTinue" - resume a paused program

Example PROG1:SEL:STAT RUN

PROGram:SELected:STATe?

Description This query returns a value indicating the status of the loaded program.

Syntax PROG(0..26):SEL:STAT?

Response

"STOPPED"	- the program is stopped
"PAUSED"	- the program is paused
"RUNNING"	- the program is in progress

Example PROG1:SEL:STAT?

7.10 IQ-9100 Switch Module Commands

PROGram[:SELected]:EXECute

Description: This command initiates execution of the program currently loaded in the module's memory.

Syntax PROG(0..26)[:SEL]:EXEC

Parameters N/A

Example PROG1:SEL:EXEC

PROGram[:SELected]:NAME

Description This command loads a previously defined switch program into the modules memory. See the module Instruction Manual for information about creating a switching program.

Syntax PROG(0..26)[:SEL]:NAME<space><string>

Parameters The <string> parameter is the path and filename of the program to be loaded

Example PROG1:SEL:NAME C:\IQ\USERFILE\PROG1x4.prg

Note This command will terminate any program that may be executing.

PROGram[:SELected]:NAME?

Description This query returns the name and path of the currently loaded program.

Syntax PROG(0..26)[:SEL]:NAME?

Response Returns a string identifying the path and name of the program file in the format;
"C:\IQ\USERFILE\PROG1x4.PRG"

Example PROG1:SEL:NAME?

PROGram[:SELected]:STATe

Description This command controls the program currently loaded in the module memory.

Syntax PROG(0..26)[:SEL]:STAT<space><value>

Parameters The <value> parameter can be:
"0" or "RUN" - start the program
"1" or "PAUSe" - pause the program
"2" or "STOP" - stop program execution
"3" or "CONTinue" - resume a paused program

Example PROG1:SEL:STAT RUN

PROGram[:SELected]:STATe?

Description This query returns a value indicating the current status of the loaded program.

Syntax PROG(0..26)[:SEL]:STAT?

Response
"STOPPED" - the program is stopped
"PAUSED" - the program is paused
"RUNNING" - the program is in progress

Example PROG1:SEL:STAT?

ROUTe:CLOSe

Description This command positions the optical switch to the reset position. In this position there is no optical continuity.

Syntax ROUT(0..26):CLOS

Parameters N/A

Example ROUT1:CLOS

ROUTe:OPEN

Description This command makes the switch change from the reset position (no optical continuity) to the channel position in effect when the switch was closed.

Syntax ROUT(0..26):OPEN

Parameters N/A

Example ROUT1:OPEN

ROUTe:OPEN?

Description This query returns a value indicating whether the switch is OPEN or CLOSED

Syntax ROUT(0..26):OPEN?

Response "1" - there is optical continuity
"0" - there is no optical continuity

Example ROUT1:OPEN?

ROUTe:PATH:CATalog?

Description This query returns a value indicating the type of switch, e.g.. 1x2, 1x4, 1x12, 1x16, 1x32, or 2x2.

Syntax ROUT(0..26):PATH:CAT?

Response The response is in the format:
"9 x 99"

Example ROUT1:PATH:CAT?

ROUTe:SCAN

Description This command sets the switch to a specific channel.

Syntax ROUT(0..26):SCAN<space><value>

Parameters The <value> parameter can be:
"1" to "32" -depending on the switch configuration
"BYPASS" - 2x2 switch only
"INSERT" - 2x2 switch only

Example ROUT1:SCAN 4

ROUTe:SCAN?

Description This query returns a value indicating the current switch position.

Syntax ROUT(0..26):SCAN?

Response
"99" - number 1 to 32 indicating the position or
"1" - in bypass position (2x2 switch only)
"2" - in insert state (2x2 switch only)

Example ROUT1:SCAN?

7.11 Quick Reference Command Trees

The following subsections contain quick reference tables illustrating the command tree for each of the IQ modules. See the detailed explanation of each command for more complete information.

7.11.1 Optical Test System

COMMAND		PARAMETER/ RESPONSE	DESCRIPTION
SHEL	LIGH	<boolean> 1|ON; 0|OFF	-control front panel lights
	LIGH?	(0|1)	-get light status
	DATE	<YYYY>,<MM>,<DD>	-set system date
	DATE?	(MM/DD/YY)	-get system date
	TIME	<HH>,<MM>,<SS>	-set system time
	TIME?	(HH:MM:SS)	-get system time
INST	CAT?	(module, module,...)	-get available modules
	FULL?	(module# -module, ...)	-get available modules with module address
	STAT	<boolean>,<value> (1|ON|0|OFF;0..26)	-open or close application
	STAT?	<value> returns(0|1) (0..26)	-is application is open

7.11.2 IQ-1100 and IQ-1200 Power Meters

COMMAND					PARAMETER/ RESPONSE	DESCRIPTION
UNIT	**POW**			*	<W|DB|DBM>	-set display units
FORM	**READ**	**[DATA]**		*	< 0 | 1 | 2| 3| 4>	-set display resolution
SENS	**AVER**	**[STAT]**		*	<0 | 1>	-set data averaging
		STAT?			(0|1)	-get average status
	CORR	**COLL**	**ZERO**	*	<[ALL]>	-null measurement
		OFFS	**[MAGN]**		<99.999> [DB]	-set an offset value
	FREQ			*	<value> [HZ]	-set sampling rate
		CATA?			(99.9,.....)	-list sampling rates
	FREQ?				(99.9)	-get sampling rate
	POW	**UNIT**		*	<DBM|W|0|1>	-set display units
		UNIT?			(0|1)	-units set to watt?
		WAVE		*	<9999> [NM]	-set wavelength
		WAVE?			(9999)	-get wavelength
		REF?			(+99.999 dBm)	-get reference value
		REF	**DISP**	*		-set new reference
			STAT	*	<0 | 1>	-set absolute or relative
			STAT?		(0 | 1)	-get absolute or relative
ABOR						-stop all measurements
INIT	**CONT**				<0 | 1>	-start continuous measurements

	CONT?			(0 | 1)	-measurements started?
	[IMM]				-store one measurement
FETC	[SCAL]	POW	DC?	(+999.999E-99)	-read stored value
READ	[SCAL]	POW	DC?	(+999.999E-99)	-store and read value
INST	NSEL			<channel#>	-set channel number
MMEM	ACQ			<1|0>	-start/stop acquisition
	ACQ?			(1|0)	-acquisition started?

* These commands are not executed if a data acquisition is in progress. The message, "Acquisition already running" will be returned.

7.11.3 IQ-2100 Source

COMMAND				PARAMETER/ RESPONSE	DESCRIPTION
SOUR	AM	[INT]	FREQ	<value> [HZ | KHZ] | CW	-set internal modulation
			FREQ?	(9999)	-read internal modulation
	POW	ATT		<value>[0..10] [DB]	-set attenuation
		ATT?		(99.9)	-read attenuation
		STAT		[value] | [ON | OFF]	-turn source ON or OFF
		STAT?		(0 | 1)	-read source state
		WAVE		UPP | LOW	-set wavelength
		WAVE?		(9999)	-read wavelength

7.11.4 IQ-3200 Return Loss Meter

COMMAND				PARAMETER/ RESPONSE	DESCRIPTION
FORM	READ	DATA		<0|1|2|3|4|>	-set display resolution
		DATA?		(0|1|2|3|4)	-get display resolution
INP	OFFS			<±99.999> [DB]	-set offset value
SENS	AVER	STAT		<1|ON | 0|OFF>	-set data averaging
		STAT?		(0|1)	-data averaging?
	CORR	COLL	ZERO		-null measurement
	POW	REFL			-ORL zero measurement
		REF		<99.999 | EXFO [DB|%]>	-ORL calibration
		WAVE	SET	<9999> [NM]	-set wavelength
			PROG	<1 | 2>,<9999> [NM]	-program wavelength
			LIST?	(9999 - 9999)	-get programmed wavelengths
		WAVE?		(9999)	-get wavelength
ABOR					-stop measurements
INIT	CONT			< 1|ON | 0|OFF>	-start/stop measurement
	CONT?			(TRUE|FALSE)	-measurements started?
	[IMM]				-store one value
FETC?				(+999.999)	-get stored value
READ	POW?			(+999.999)	-store and get value

7.11.5 IQ-3100 Variable Attenuator / IQ-3300 Variable Reflector

COMMAND			PARAMETER/ RESPONSE	DESCRIPTION
INP	ARES?		(-9.999)	-get max resolution
	ATT		[1] <99.999 [DB] | MIN|MAX>	-set attenuation
	ATT?		(± 999.999)	-get attenuation
		MAX	(- 999.999)	-get max attenuation
		MIN	(- 999.999)	-get min attenuation
	OFFS		[1] <±999.999 [DB]>	-set an offset value
		DISP		-set offset to display
	OFFS?		(±999.999)	-read offset value
	WAV		[1] <9999.9 [NM]>	-set wavelength
	WAV?		(9999.9)	-get wavelength
		MAX	(9999.9)	-get max wavelength
		MIN	(9999.9)	-get min wavelength
OUTP	APM		<OFF|ON|0|1>	-set absolute or relative
	APM?		(0|1)	-is absolute display?
	STAT		[2] <OFF|ON|0|1>	-activate shutter
	STAT?		(0|1)	-get shutter state
PROG	SEL	NAME	<string>	-load a program
		NAME?	(string)	-get program name
		STAT	<0..3 | RUN| PAUS | STOP | CONT>	-control the program
		STAT?	(STOPPED | PAUSED | RUNNING)	-get program state
		EXEC		-execute program

[1] Valid parameters are determined by the type of module.
[2] The file name must include the drive and the complete path.

7.11.6 IQ-9100 Optical Switches

COMMAND			PARAMETER/ RESPONSE	DESCRIPTION
ROUT	PATH	CAT?	(9 X 99)	-get type of switch
	CLOS			-turn off switch
	OPEN			-turn on switch
	OPEN?		(1 | 2)	-is switch on?
	SCAN		<1..32 | BYPASS | INSERT>	-set channel
	SCAN?		(99)	-get current channel
PROG	[SEL]	NAME	<string>	-load a program
		NAME?	(string)	-get program name
		STAT	<0..3 | RUN| PAUS | STOP | CONT>	-control the program
		STAT?	(STOPPED | PAUSED | RUNNING)	-get program state
		EXEC		-execute program

8 ERROR MESSAGES

8.1 Error Message Format

System and device specific errors are managed by the SCPI Manager. The generic format for error messages is illustrated in Figure 8-1.

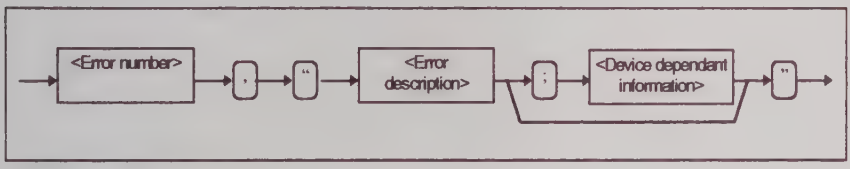

Figure 8-1. Error Message Format

As shown in the figure, the message contains three parts; the error number, the error description, and device dependent information. All error messages are stacked in a FIFO buffer. When there is at least one message in the buffer, bit 2 of the Status Byte Register is set to 1. Use the command **SYST:ERR?** to read the most recent message. The error message buffer is initialized when starting the SCPI Manager, when executing the command *CLS, or by reading the last message stored in the buffer.

- Error messages ending in a negative number are SCPI based errors.
- Error messages ending in a positive number are specific to the IQ-200 system components

8.2 SCPI Manager Errors

ERROR NUMBER	DESCRIPTION	PROBABLE CAUSE
-100,	"Command Error"	- the SCPI manager does not recognize the command, likely due to incorrect grammar
-104,	"Data Type Error"	- a data parameter is not the expected data type
-108,	"Parameter not allowed"	- too many data parameters were detected for the specified command
-109,	"Missing Parameter"	- a required data parameter was not detected
-113,	"Undefined Header"	- a module address number was missing from the command
-200,	"Execution Error"	- the module addressed does not support this command - a device error
-224,	"Ilegal parameter value"	- a parameter was outside the expected range
-300,	"Device Dependent Error"	- possibly a hardware problem
-400,	"Query Error"	- attempt to read an empty buffer - buffer data has been lost

8.3 IQ-200 Optical Test System Error Messages

ERROR NUMBER	DESCRIPTION	PROBABLE CAUSE
2,	"Application error."	- the system was unable to load the requested module application

8.4 IQ-1100, IQ-1200, and IQ-1500 Power Meter Error Messages

ERROR NUMBER	DESCRIPTION	PROBABLE CAUSE
101,	"Unknown Command."	- the power meter has received a command that it does not recognize
102,	"Not compatible with current channel."	- the command is not supported by the IQ-1200 4-Channel Power Meter
103,	"Light on detector."	- a null command has been attempted with light reaching the detector
104,	"Unknown parameter."	- the power meter has received a data parameter that it was not expecting
105,	"Parameter out of range."	- the power meter has received a data parameter outside the valid range
106,	"Acquisition already running."	- command was sent to start data acquisition while acquisition was already in progress
107,	"Acquisition not running."	- a command to stop acquisition was sent with no acquisition in progress
108,	"Cannot open file."	- available disk space is less than 10 Mbytes when attempting to perform a data acquisition
109,	"No data."	- a FETCh command was received without a previous INIT, i.e. there is no data in the data output buffer

8.5 IQ-2100 Source Error Messages

ERROR NUMBERS	DESCRIPTION	PROBABLE CAUSE
402,	"Ilegal parameter value."	- an illegal data parameter has been received
403,	"Attenuation out of range."	- a command has attempted to set the internal attenuation to a an unsupported value

8.6 IQ-3200 Return Loss Meter Error Messages

ERROR NUMBER	DESCRIPTION	PROBABLE CAUSE
301,	"Unknown command."	- the IQ-3200 has received a command that it does not recognize
302,	"Light on detector."	- a null command has been attempted with light reaching the detector
303,	"Incorrect reference."	- an invalid calibration reference value has been specified
304,	"Error"	- a wavelength selection error has occurred

8.7 IQ-3100 and IQ-3300 Error Messages

ERROR NUMBER	DESCRIPTION	PROBABLE CAUSE
201,	"Unknown command."	- the module has received a command that it does not recognize
202,	"Communication error."	- a command has taken longer than expected to complete execution
203,	"Settings conflict."	- there is a conflict between the current state of the program and the program control command just received
204,	"Command error."	- an invalid data parameter has been received
205,	"Wavelength list full."	- a command has attempted to add more than 100 wavelengths to the wavelength list box
206,	"No program loaded."	- an attempt has been made to activatea program, but no such program is loaded into the module memory
207,	"Program running."	- a command that will halt the currently executing program has been received. An attempt has been made to override a program setting.

8.8 IQ-9100 Switch Error Messages

ERROR NUMBER	DESCRIPTION	PROBABLE CAUSE
502,	"Communication error."	- a command has taken longer than expected to complete execution
503,	"Illegal program name."	- an illegal or nonexistent path or filename has been received
504,	"Settings conflict."	- there is a conflict between the current state of the program and the program control command just received
505,	"Channel out of range."	- an invalid channel number has been requested
506,	"Program running."	- a command has been refused because a program is running
507,	"Illegal parameter value."	- an invalid data parameter has been received

9 GENERAL INFORMATION

9.1 Warranty

EXFO Electro-Optical Engineering Inc. (EXFO) warrants this equipment against defects in material and workmanship for a period of one year from the date of original shipment. EXFO also warrants that this equipment will meet applicable specifications under normal use.

During the warranty period, EXFO will, at its option, repair, replace, or issue credit for any defective product. This warranty also covers recalibration during the first year if the equipment is repaired or if the original calibration is erroneous.

IMPORTANT

The warranty can become null and void if:

- ***the equipment has been tampered with, repaired, or worked upon by unauthorized individuals or non-EXFO personnel,***
- ***the warranty sticker has been removed,***
- ***the case screws have been removed,***
- ***the case has been opened,***
- ***the equipment serial number has been altered, erased or removed, or***
- ***the equipment has been misused, neglected or damaged by accident.***

THIS WARRANTY IS IN LIEU OF ALL OTHER WARRANTIES EXPRESSED, IMPLIED OR STATUTORY, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL

EXFO BE LIABLE FOR SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES.

9.2 Liability

EXFO shall not be liable for damages resulting from the use of the purchased product, nor shall EXFO be responsible for any failure in the performance of other items to which the purchased product is connected, or the operation of any system of which the purchased product may be a part.

9.3 Exclusions

EXFO reserves the right to make changes in the design or construction of any of its products at any time, without incurring any obligation to make these changes on units previously purchased. Accessories — including but not limited to fuses, pilot lamps and batteries — used with EXFO's products are not covered by this warranty.

9.4 Certification

EXFO certifies that this equipment met its published specifications at the time of shipment from the factory. EXFO further certifies that its calibration measurements are traceable to the U.S. National Institute of Standards and Technology (NIST) laboratories. If this equipment requires regular recalibration, a Certificate of Calibration will be included with the unit, replacing the standard Certificate of Quality. The Certificate of Calibration contains the date of the equipment's last calibration and the recommended interval for recalibration.

9.5 Service and Repairs (Under Warranty or Other)

To obtain service or repair for any equipment, follow the procedure below.

- Call the EXFO Sales and Product Support Group. Support personnel will determine if the equipment requires service, repair or calibration.

Sales and Product Support Group:

USA and Canada

Tel.: 1-800-663-EXFO
1-800-663-3936
Fax: 1-418-683-2170
E-Mail: exfo@riq.qc.ca

International

Tel.: 1-418-683-0211
Fax: 1-418-683-2170
E-Mail: exfo@riq.qc.ca

EXFO Europe

Tel.: 33-1-34.63.00.20
Fax: 33-1-34.65.90.93
E-Mail: exfoeur@ibm.net

- If the equipment must be returned to EXFO or an authorized service center, support personnel will issue a Return Merchandise Authorization (RMA) number and an address for return.
- Pack the equipment in its original shipping material. Be sure to include a statement or report fully detailing the defect and the conditions under which it was observed.

IMPORTANT

Never send any unit or accessory back to EXFO without a Return Merchandise Authorization (RMA) number.

- Return the equipment, prepaid, to the address given by the support personnel. Be sure to write the RMA number on the shipping slip. EXFO will return any package which does not bear an RMA number.

Note: *A test setup fee will be charged for any returned unit which, upon testing, is found to meet the applicable specifications.*

- After repair, the equipment will be returned with a repair report. If the equipment is not under warranty, the customer will be invoiced for the cost appearing on this report. EXFO will cover the cost of returning equipment under warranty to the customer. However, shipping insurance is at the customer's expense.

10 INDEX

A

Acquisition
  data 7-25, 7-68
Active connections 3-4
Address
  default 3-7
  GPIB 3-4, 3-6
Applications 2-7, 5-1
  client 5-3
  DDE linked 3-7
  developing 5-7
  high level 3-2
  IQ 5-2
  modules 3-5
  single software 3-1

B

Bidirectional data sharing 2-3

C

Client
  destination 5-1
Code
  source 5-8
Commands 5-1, 7-14
  Optical test system 7-14
  power meter module 7-21
Commands, 9100-Switch module
  ROUT SCAN 7-64
Commands, common
  CLS 7-4
  ESE 7-4, 7-5
  ESR 7-5
  IDN 7-5
  OPC 7-6
  OPC? 7-6
  RST 7-7
  SRE 7-7
  SRE? 7-8
  STB? 7-8
  TRG 7-8
  TST? 7-9
  WAI 7-9
Commands, IQ Power meter module
  ABOR 7-21
  FETC [SCAL] POW DC? 7-22
  FORM READ[DATA] 7-22
  INIT CONT? 7-23
  INIT CONT 7-23
  INIT [IMM] 7-24
  INST NSEL 7-24
  MMEM ACQ 7-25
  MMEM ACQ? 7-25
  READ [SCAL]POW DC? 7-26
  SENS AVER STAT? 7-27
  SENS AVER[STAT] 7-26
  SENS CORR COLL ZERO 7-27
  SENS CORR OFFS[MAGN] 7-28
  SENS FREQ 7-29
  SENS FREQ CATA? 7-29
  SENS FREQ? 7-30

SENS POW REF DISP....7-30
SENS POW REF STAT....7-31
SENS POW REF STAT?....7-31
SENS POW REF?....7-32
SENS POW UNIT....7-32
SENS POW UNIT?....7-33
SENS POW WAVE....7-33
SENS POW WAVE?....7-34
UNIT POW....7-34
Commands, IQ-200 Optical test system
INST\ CAT\ FULL?....7-15
INST CAT?....7-14
INST STAT....7-16
INST STAT?....7-17
SHEL DATE....7-17
SHEL DATE?....7-18
SHEL\ LIGH....7-18
SHEL LIGH?....7-19
SHEL TIME....7-19
SHEL TIME?....7-20
Commands, IQ-2100 source
SOUR AM[INT] FREQ?....7-45
SOUR AM[INT] FREQ....7-45
SOUR POW ATT....7-46
SOUR POW ATT?....7-46
SOUR POW STAT....7-47
SOUR POW STAT?....7-47
SOUR POW WAVE....7-48
Commands, IQ-3100, IQ-3300
INP ARES?....7-49
INP ATT....7-50
INP ATT?....7-51
INP OFFS....7-51
INP OFFS DISP....7-52
INP OFFS?....7-52
INP WAV....7-53
INP WAV?....7-54
OUTP APM?....7-55
OUTP STAT?....7-56
OUTP[STAT]....7-56
PROG SEL EXEC....7-57
PROG SEL NAME....7-57
PROG SEL NAME?....7-58
PROG SEL STAT....7-58
PROG SEL STAT?....7-59
Commands, IQ-3200 return loss meter
ABOR....7-35
FETC?....7-35
FORM READ DATA....7-36
FORM READ DATA?....7-36
INIT [IMM]....7-38
INIT CONT....7-37
INIT CONT?....7-37
INP OFFS....7-38
READ POW?....7-39
SENS AVER STAT?....7-40
SENS AVER STAT....7-39
SENS CORR COLL ZERO....7-40
SENS POW REF....7-41
SENS POW REFL....7-42
SENS POW WAVE LIST?....7-42
SENS POW WAVE PROG....7-43
SENS POW WAVE SET....7-44
SENS POW WAVE?....7-44
Commands, IQ-9100 Switch module
PROG[SEL] EXEC....7-60
PROG[SEL] NAME....7-60
PROG[SEL] NAME?....7-61
PROG[SEL] STAT....7-61
PROG[SEL] STAT?....7-62
ROUT CLOS....7-62
ROUT OPEN....7-63
ROUT OPEN?....7-63
ROUT PATH
CAT?....7-64, 7-64
ROUT SCAN?....7-65, 7-65
Commands, system
SYST ERR?....7-10

SYST VERS **7-10**
Commands. IQ-3100, IQ-3300
OUTP APM **7-55**
Commercial programming languages 4-4
Communication link
SCPI Manager 3-2
Compatible instrument
GPIB 2-3
Compatible instruments
GPIB 2-5
Components, IQ
remote control of 4-4
Connections
active 3-4
Controller card 2-3
GPIB 2-3, 2-5
Controller interface
GPIB 2-9
Customizing
modules 5-1

D

Data
exchange 5-1
Data acquisition 7-25, 7-68, 8-4
Data sharing
bidirectional 2-3
DDE 2-9
client 5-1, 5-3
conversation 5-3
Dynamic data exchange 1-4
link 2-4, 2-7, 2-9, 3-1, 5-2, 5-3, 5-5, 5-8
linked application 3-7
protocol 5-2
server 5-1, 5-3
DDE link 5-2, 5-3
Debugging 3-8
Default address 3-7
Default values 4-3
Destination
client 5-1
Destination link 5-3
Developing
integrated application 5-2
Development environments
GPIB 4-4
Development kits 2-7
GPIB 4-3
Device card
IQ-203 2-3
Device configuration
IQ-203 GPIB 4-1
DLL
Dynamic link library 1-4
DMA
Direct memory addressing 1-4
Drivers
interface, GPIB 4-2
software 2-3
virtual instrument 2-4

E

ESE
Standard event enable register 7-1
ESR
Standard event register 7-1
Exchange data 5-1
Expansion card 5-2
GPIB 2-1
IQ-206 5-5
Expansion units
IQ-206 7-11

F

Fiber-optic test system

modular 2-1
FIFO
First in first out 1-4

G

GPIB
address 3-4, 3-6
compatible instrument 2-3
compatible instruments 2-5
controller card 2-5
controller interface 2-9
controller interface card 6-1
development environments 4-4
development kits 4-3, 6-2
device configuration 4-1
General purpose bus 1-4
interface card 6-2
interface driver 4-2

H

Host computer....2-7, 2-9, 4-2, 4-3, 5-2, 5-5, 5-8, 7-12

I

IEEE 488.2 (1992) standard 4-1
Integrated IQ-applications 1-2, 6-1
creation of, ... 2-4
programming 5-4
Visual Basic 5-2
Interface card 2-9, 6-2
IQ
applications 5-2, 5-3, 5-4
components 3-1
integrated application 2-9, 5-2, 5-3
Operating system 3-2
Optical test system 7-11, 7-14
platform 2-1
virtual instrument drivers 4-4
IQ modules
control of 3-2
IQ-1100 Power Meter 7-11
IQ-200
components 1-2, 1-3, 2-5, 6-1
maximum configuration 7-12
modules 3-1
Optical test system 3-2, 3-4, 8-3
software 5-8
system configuration 1-2
IQ-200 Optical Test System 3-2, 3-4, 8-3
IQ-200 system 8-1
components 8-1
IQ-203 ..... 2-1, 5-2, 5-5, 5-8, 6-1, 7-11, 7-14, 7-15, 7-16
device card 2-3
device configuration 4-1
GPIB controller option 6-1
initiating GPIB communication with, ... 4-2
mainframe 2-3, 2-5
remote control of, ... 4-2
remote instructions 4-3
IQ-203 Controller configuration 2-5
IQ-206 ..... 2-1, 5-5
expansion units 2-9, 7-11
PC expansion card 5-5
IQ-3100
commands 7-49
IQ-3300
commands 7-49

K

Keywords 7-12

L

LED push-button 3-5
Link Item 5-3
Link Mode 5-4
Link Topic 5-3
LinkNotify 5-6
Local mode 3-7
Local Lockout 3-7
Lockout
  local 3-7
  remote 3-7
Lockout mode
  enabling 5-7

M

Mainframe
  IQ-203 ..... 2-3, 2-5
Measurement display resolution 7-11
Modes
  link 5-4
  local 3-7
  remote 3-7
Modular fiber-optic test system 2-1
Modules 2-1, 3-4, 7-11
  address modifier 7-11
  applications 3-5
  control software 2-3, 2-4
  customizing of, ... 5-1
  IQ 6-2
  IQ, control of, ... 3-2
  IQ-200 7-12
  IQ-206 expansion unit 2-9
  position 5-6
  software control 5-1
  specifying 7-11
  stand-alone instrument 2-3

O

Optical Test System
  commands 7-14
  IQ 5-8, *7-11*, 7-14
  IQ-200 3-4

P

Parameters
  SCPI 3-5
Power Meter
  IQ-1100 7-11
Programming environments 6-2
Programming languages 2-7, 6-2
  commercial 4-4
  DDE 5-2

R

Registers 7-1
  ESR 7-1
  intializing 5-5
  SRE 7-1
  STB 7-1
Remote control
  IQ components 4-4
Remote Lockout 3-7
Remote mode 3-7
Remote instructions
  IQ-203 4-3
Repair 9-4
RWLS
  remote with lockout state 3-7

S

SCPI
  parameters 3-5

Standard commands for programmable instruments 1-4
SCPI Manager 5-1, 5-5
communication link 3-2
DDE server 5-1
important concepts 3-2
main functions 3-1
registers 5-5
SCPIManInBox **5-5**
SENS(0..26):FREQ? 7-30
Serial poll 7-3
Server 5-1
Service request 7-1
Software
drivers 2-3
module control 5-1
Source 5-1
code 5-8
SRE
Service request enable register 7-1
stand-alone instruments 2-1
STB
Status byte resgister 7-1

T

Topic
link 5-3
Troubleshooting 3-8

V

Values
default 4-3
Virtual instrument drivers 2-4
Visual Basic
integrated applications 5-2

W

Windows Dynamic Data Exchange 3-1

CORPORATE HEADQUARTERS
465 Godin Avenue
Vanier, QC, Canada
G1M 3G7
Tel.: 1-800-663-3936
(418) 683-0211
Fax: (418) 683-2170

info@exfo.com
www.exfo.com

www.manualsplus.com
1025855

Printed in Canada